Analysis and Detection of Tap Changes on Power Systems



Xiaoqing Chen

St Cross College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity Term 2018

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Prof. Stephen Duncan. Throughout my doctoral studies, your insights, encouragement, and advice have been invaluable. Without your support and guidance, it would be impossible for me to accomplish my research project and achieve my D.Phil degree. I am also very grateful to Prof. Manus Henry and Dr. Ross Drummond, whose generous help greatly contributed to my research.

Also, I would like to thank the China Oxford Scholarship Fund for its financial support during my studies. It has been a great honour for me to become a China Oxford Scholar.

I feel truly privileged to have pursued my doctoral research within the Control Group at the University of Oxford, where I had the opportunity to work alongside many talented and brilliant researchers and students - Ross, Manuel, Mehdi, Ming, Bartolomeo, Goran, Yang, Reza, Dhruva, Andreas, John, Shuhao, Xiaonan. I will always treasure the stimulating discussions, inspiring conversations, and joyful moments we shared together.

I am also thankful for the chance to meet and connect with students and scholars from diverse disciplines at Oxford - Yixing, Yicong, Xiaochun, Yang, Yifan, Haoyu, Zhu, Xue. Our friendship has enriched my life here and made my time in Oxford both colourful and memorable.

I also want to say thank you to Xizi Wu for always being supportive and believing in me.

Finally, I owe my deepest thanks to my family, especially to mother, Feng Yu, for her unwavering support. It was her encouragement that motivated me to explore this big world, from my hometown Jinzhou to Tsinghua University in Beijing, and then to the University of Oxford, where I was able to complete my D.Phil.

Abstract

Demand side management (DSM) is a key concept for the successful implementation of smart grids. In order to achieve DSM, the users need information about local network stress and one approach to detecting network stress is to monitor the on-load tap changers (OLTC) within the network, since tap changes play an important role in voltage regulation in power system and can reflect the variation of load. This thesis focuses on the analysis, modelling and detection of tap changes.

After an introduction on power systems and the OLTC transformer, an algorithm is proposed that can be used for measuring the frequency of the three phase sinusoidal voltage signal in power system. The frequency estimation algorithm contains two parts. The first part is a HTPLL or filtering algorithm that uses the Prism methodology, which provides an initial frequency estimate. The second part is a complex least mean square (CLMS) algorithm that can refine the initial frequency estimate. The proposed frequency estimation algorithm is tested with both simulated and real data and compared with other algorithms, showing that the proposed algorithm can give improved performance.

In the third chapter, an algorithm for tap change detection is introduced. The algorithm is divided into three parts. The first part is a Hilbert-transform phase-locked loop (HTPLL), which is used to estimate the frequency of the voltage signal. The second part is a QR-decomposition windowed recursive least squares algorithm, which is used to estimate the amplitude of the voltage signal. The last part is a hypothesis test that can distinguish tap changes from the variation in the voltage magnitude from the voltage fluctuation caused by changes in the load. The algorithm is tested with both simulated and real data, showing that the algorithm is capable of detecting tap changes from the voltage signal.

In the fourth chapter, models of tap changer, generator and load buses are introduced which can be used for studying the interaction between tap changes and variations in the load. We investigate different tap changer modelling methods and propose a quantised tap changer model, which combines a linear dynamic function with a quantiser. This model evolves in continuous time but has a quantised number of tap positions. The response of different tap changer models are simulated, demonstrating that the quantised tap changer is suitable for power system analysis.

Contents

List of Figures vi

1 Introduction 1

1.1 The structure of a power system . . . 2
1.2 Variability of the supply and demand of power system . . . 4
1.2.1 The balance in power system . . . 4
1.2.2 Variation at supply and demand side . . . 6
1.2.3 The concept of demand side management . . . 8
1.3 The control layers of power system . . . 11
1.4 OLTC transformer and tap change . . . 13
1.4.1 OLTC transformer structure . . . 15
1.4.2 Tap changes . . . 16
1.5 Outline of thesis . . . 17
1.6 Contributions . . . 19

2 Frequency estimation of power system 21

2.1 Introduction . . . 21
2.2 Structure of the algorithm . . . 23
2.3 Initial estimation method . . . 24
2.3.1 Phase-locked loop . . . 24
2.3.2 Prism algorithm . . . 33
2.4 CLMS algorithm for result enhancement . . . 35
2.4.1 Fixed step CLMS enhancement method . . . 35
2.4.2 Adaptive step CLMS enhancement method . . . 38
2.5 The combined frequency estimation algorithm . . . 40
2.5.1 Fixed step CLMS enhancement method . . . 40
2.5.2 Adaptive step CLMS enhancement method . . . 42
2.6 Simulation and Discussion . . . 44
2.6.1 Fixed step CLMS enhanced HTPLL algorithm . . . 44
2.6.2 Fixed step CLMS enhanced Prism algorithm . . . 50
2.6.3 Adaptive step CLMS enhanced HTPLL algorithm . . . 56
2.6.4 Adaptive step CLMS enhanced Prism algorithm . . . 63
2.7 Conclusion . . . 69

3 Automatic detection of tap changes **71**
3.1 Introduction . . . 71
3.2 Algorithm structure . . . 73
3.3 Frequency tracking technique . . . 75
3.3.1 Classical PLL . . . 75
3.3.2 Hilbert-transform PLL . . . 80
3.4 Recursive LMS algorithm . . . 83
3.4.1 Classical recursive least square algorithm . . . 84
3.4.2 QR decomposition windowed recursive least square algorithm 90
3.5 Hypothesis test . . . 94
3.5.1 Generalized likelihood ratio test . . . 95
3.5.2 Two sample t-test . . . 99
3.6 Simulation and discussion . . . 102
3.6.1 PLL result with real input . . . 102
3.6.2 QRD-WRLS algorithm result with real input . . . 104
3.6.3 Hypothesis test result . . . 109
3.6.4 GLR test result . . . 110
3.6.5 Two sample t-test result . . . 114
3.7 Conclusion . . . 117

4 Modelling of tap changer and power system **120**
4.1 Introduction . . . 121
4.2 Power system models . . . 122
4.2.1 Finite generator bus model . . . 122
4.2.2 Infinite generator bus model . . . 126
4.2.3 ODE model of the infinite generator bus model . . . 127
4.3 Quantized tap changer model . . . 129
4.3.1 Discrete tap changer model . . . 129
4.3.2 Continuous tap changer model . . . 130
4.3.3 Quantised tap changer model . . . 130
4.4 Comparison of different types of models . . . 131
4.4.1 Finite generator bus model combined with continuous & quantised tap changer models . . . 132
4.4.2 Infinite generator bus model . . . 134
4.5 Removing OLTC chattering using a deadzone . . . 136
4.5.1 Finite generator bus model . . . 137
4.5.2 Infinite generator bus model . . . 140
4.6 Simulation of a 6-bus system . . . 141
4.7 Conclusion . . . 146

5 Conclusion **147**

5.1 Summary and conclusion 147

5.2 Future work 151

References **152**

List of Figures

1.1 Diagram of an electric power system [6] . . . 3
1.2 Structure of a conventional electric power system [7] . . . 4
1.3 Frequency variation over an hour on the 8th of May 2018 . . . 5
1.4 Frequency variation over 24 hours on the 12th of March 2014 . . . 6
1.5 Average total power system demand [25] . . . 8
1.6 Load shaping techniques of DSM [31] . . . 9
1.7 Time-scale decomposition of controls and actions in the power grid [51] 12
1.8 An OLTC transformer . . . 14
1.9 The diagram of an OLTC transformer . . . 15
1.10 Signal of a real tap change . . . 17
1.11 Signal of a real tap change (detail of the top portion highlighted by the red rectangular in Figure 1.10) . . . 18

2.1 The diagram of complex LMS enhanced HTPLL algorithm for frequency estimation in three-phase power system . . . 24
2.2 Diagram of PLL. The phase detector, low-pass loop filter and voltage-controlled oscillator are connected together to form a PLL, which is a close-loop system. . . . 25
2.3 Diagram of a HTPLL [96] . . . 29
2.4 Comparison of the performance of HTPLLs with different ω_n . . . 33
2.5 The response of Prism algorithm to a step change in input frequency 35
2.6 Diagram of the fixed step CLMS enhanced frequency estimation algorithm . . . 40
2.7 Diagram of the adaptive step CLMS enhanced frequency estimation algorithm . . . 43
2.8 Performance of different frequency estimation techniques with a step change in input frequency . . . 45
2.9 Performance of different frequency estimation techniques with changing input frequency . . . 47
2.10 The frequency estimation error of different techniques . . . 48
2.11 Real voltage data . . . 49

2.12 Performance of different frequency estimation techniques with real data input . . . 50
2.13 Detail of the performance of different frequency estimation techniques with real data input (detail of Figure 2.12) . . . 51
2.14 Performance of different frequency estimation techniques with a step change in input frequency . . . 52
2.15 Performance of different frequency estimation techniques with changing input frequency . . . 54
2.16 The frequency estimation error of different techniques . . . 55
2.17 Performance of different frequency estimation techniques with real input data (the small spikes in the figure is caused by printing issue) 56
2.18 Performance of different frequency estimation techniques with step change input . . . 57
2.19 Changing of the square of absolute value of the complex error term ζ 58
2.20 Variation of μ_k during the step change . . . 59
2.21 Performance of different frequency estimation techniques with changing input frequency . . . 60
2.22 The frequency estimation error of different techniques . . . 61
2.23 The variation of $\bar{\zeta}\zeta$ and μ_k . . . 62
2.24 Performance of different frequency estimation techniques with real data input . . . 62
2.25 Performance of different frequency estimation techniques with step change input . . . 64
2.26 The variation of $\bar{\zeta}\zeta$ and μ_k . . . 65
2.27 Performance of different frequency estimation techniques with changing input frequency . . . 66
2.28 The frequency estimation error of different techniques . . . 66
2.29 The variation of $\bar{\zeta}\zeta$ and μ_k . . . 67
2.30 Performance of different frequency estimation techniques with real data input . . . 68
2.31 Detail of the performance of different frequency estimation techniques with real data input shown in Figure 2.30 . . . 68

3.1 Algorithm structure of the tap change detection diagram . . . 74
3.2 The responses of PLLs to a step change. The PLLs has the same damping factors of $\zeta = 1$ and different natural frequencies. Comparing the results, it can be seen that when the natural frequency is smaller, the transient time for the PLL to react to the input step change is longer but the amplitude of the second order harmonic in its output is smaller. . . . 79

3.3 Classical PLL output in lock-in condition. The output is fluctuating around the input at twice the input frequency. This fluctuation is caused by the second order harmonic in the PD output. 80
3.4 Classical PLL response to input frequency change. The input step change occurs at $t = 5$s. The PLL output shows a transient process and locked-in to the input in about 0.5s. 81
3.5 Classical PLL response to changing input frequency 82
3.6 HTPLL response to a step input frequency change. As the HTPLL uses a HTPD which dose not generate a second order harmonic, output of HTPLL is clearer than that of the classical PLL 83
3.7 HTPLL reaction to input frequency fluctuation 84
3.8 RLS algorithm result for amplitude calculation. As the RLS algorithm takes all of the data into calculation, it reacts to input step change slowly. 87
3.9 WRLS algorithm result for amplitude calculation. There is a step change in the input magnitude at $t = 0.5$s. After the change occurs, the WRLS algorithm reacts to the step change quickly. 89
3.10 QRD-WRLS algorithm result for amplitude calculation 93
3.11 Signal of a real tap change (only the top portion of the sinusoidal voltage signal is shown) 94
3.12 GLR result for change detection. When the change occurs at $t =$ 1s, the GLR statistics increases and exceeds the threshold quickly, whereas when there is no change, the result is comparatively small. 99
3.13 GLR result for change detection, which shows the detail around the change time point in Figure 3.12 100
3.14 Two sample t-test result for change detection. There is a change in the input at $t = 1$s, and the two sample t-test statistic appears a peak when the change occurs. 102
3.15 Section of two sample t-test result for change detection in Figure 3.14, showing region around change point 103
3.16 Traditional PLL and HTPLL output frequency for the same real input data. The classical PLL output is fluctuating strongly due to the second order harmonic contents whereas the HTPLL output is clearer. 104
3.17 Diagram of the combined algorithm of HTPLL and QRD-WRLS . . 105
3.18 HTPLL and QRD-WRLS sliding window 106
3.19 HTPLL and QRD-WRLS for amplitude calculation. The blue curve shows the top portion of the sinusoidal input voltage, and the red curve shows the combined algorithm output. The combined algorithm provides an estimation of the magnitude of the input signal. 107

3.20 FFT result of the input voltage data in 1 second. 108
3.21 Detail of the amplitude of high harmonic contents in Figure 3.20 . . 108
3.22 QRD-WRLS output including up to 25th harmonics, which provides a good estimate on the input voltage magnitude. 109
3.23 Sliding window movement order 110
3.24 GLR result with real tap change input. When the tap change occurs at around $t = 125.6$, the GLR statistic exceeds the threshold quickly. 112
3.25 The magnitude variation of a voltage fluctuation 114
3.26 GLR curve of the voltage fluctuation, which shows a different pattern from the GLR curve of a tap change 115
3.27 Sliding windows for two sample t-test 116
3.28 Two sample t-test result for tap change 117
3.29 Two sample t-test curve for voltage fluctuation 119

4.1 Tap change occuring within in a power network. This figure shows the the upper peak of the sinusoidal voltage signal recorded at the load side with the jump due to a tap change occurring. This information can be used to detect network stress at the user side. 122
4.2 System structure of the finite generator bus model. The model can be divided into two subsystems. Subsystem I is the frequency control section and Subsystem II describes the tap changer and voltage dynamics [59]. 124
4.3 Diagram of a simplified power system network with a single infinite generator bus. In this simplified system, there is only one generator bus and one load bus, which are connected to each other via a power transmission system equipped with OLTC transformer. 126
4.4 Diagram of a single finite generator bus and single load bus system of the finite generator bus model. The OLTC transformer is considered to be located within the load bus. 133
4.5 Simulation of OLTC transformer ratio n, load bus voltage V and active load power P for finite generator bus model of (4.1) and (4.2) with quantised tap changer model (4.16). The output of the system shows a pattern of chattering around its equilibrium point. 133
4.6 Simulation result of a single infinite generator bus model shown in Figure 4.3 connected with the discrete tap changer model of (4.13). The variation of the OLTC transformer ratio n, load bus voltage V, active load status x_p and active power P_d show a pattern of limit cycle. In the figure, V, x_p and P_d result are in per unit (p.u.). . . . 135

4.7 Simulation result of a single infinite generator bus model shown in Figure 4.3 connected with the discrete tap changer model of (4.16). The variation of the OLTC transformer ratio n, load bus voltage V, active load status x_p and active power P_d show a pattern of chattering around the equilibrium point. In the figure, V, x_p and P_d result are in per unit (p.u.). 136
4.8 Detail of the chattering shown in Figure 4.7 137
4.9 Diagram of the Subsystem II of the finite generator bus model from Figure 4.2 with the deadzone chattering removal method included. In this case, the OLTC transformer dynamics is first processed by a deadzone and then put into an integrator and quantiser. 137
4.10 Simulation of the quantised tap change model with a well designed deadzone interconnected within the single bus finite generator network model. By introducing the deadzone, the chattering problem is solved. The transformer ratio n stays at the expected quantisation interval. The load bus voltage V and active load power P results get rid of the chattering problem as well. 140
4.11 Simulation result of a single infinite generator bus model shown in Figure 4.3 connected with the discrete tap changer model with deadzone. With the deadzone, the chattering problem is solved and the transformer ratio stays at the expected quantisation interval. . . 141
4.12 A 6-bus network [59]. *Bus* 1 and *Bus* 2 are two generator buses and *Bus* 3 to 6 are four load buses. Each load bus is equipped with OLTC transformers and are interconnected with other load buses. . 142
4.13 Simulation of the OLTC transformer ratio for the 6-bus network of Figure 4.12. The variation of OLTC transformer ratio in each load bus corresponds to the change in reactive power demand. 143
4.14 The voltage for the simulation of the 6-bus network of Figure 4.12. When the OLTC transformer ratio at each load bus changes, the voltage also changes. 144
4.15 Simulation showing the operating frequency deviation of the 6-bus network model. When the reactive power demand at a load bus changes, the operating frequency of the connecting buses fluctuates. The fluctuations are always within the acceptable range. 145

1 Introduction

Contents

1.1 The structure of a power system 2
1.2 Variability of the supply and demand of power system 4
1.2.1 The balance in power system 4
1.2.2 Variation at supply and demand side 6
1.2.3 The concept of demand side management 8
1.3 The control layers of power system 11
1.4 OLTC transformer and tap change 13
1.4.1 OLTC transformer structure 15
1.4.2 Tap changes 16
1.5 Outline of thesis 17
1.6 Contributions 19

The power system, which includes all the electrical components between generation and user, is responsible for supplying electrical energy. According to an estimate made by the International Energy Agency (IEA), by 2022, global domestic energy demand is expected to increase by 12% compared to 2010 levels[1]. At the same time, the power network capacity is only expected to grow by 6% [2, 3]. This points to a future power deficit with supply not able to match demand, unless the network can be redesigned to operate more effectively. The nature of the power system is also changing, with the increased adoption of renewable energy technologies introducing uncertainty (for example, due to wind gusts and clouds

etc.) which is already impacting grid stability and efficiency [4]. With increasing variability of supply and demand, the power grid is facing new challenges.

Within the broad field of research on power system network being undertaken worldwide to address these issues, this thesis focuses on one key component of the grid namely tap changes. They are used to manipulated the demand side voltage in power system, and the analysis and detection of tap changes are studied in this thesis. The main results are the development of the tap change detection algorithm and modelling technique of tap changer. This research will help to improve our understanding of power systems and thus help to deal with the imbalance between supply and demand in the future.

1.1 The structure of a power system

An electric power system is a network of electrical components that supply, transmit and use electric power. In this system, electricity is generated from various energy sources, such as fossil fuels, nuclear power and renewable energy. It can be divided into three parts:

- 1.The generators supplying power to the whole system.
- 2.The transmission system carrying the power from the generating centres to the load centres.
- 3.The distribution system delivering the power locally to nearby homes and industrial users.

The structure of an electric power system is shown in Figure 1.1. The majority of power systems rely on three-phase AC power, which is the standard for large scale power transmission and distribution systems across the world. The generators and loads are operating at almost the same frequency which is about 50Hz in countries including the United Kingdom, China, etc. or about 60Hz in USA, Japan, Brazil, etc [5]. Since the power loss is proportional to the square of the current, the electrical

energy is carried by a power transmission system to the distribution substation with a high voltage (and hence low current) in order to decrease energy loss.

When the three-phase power is delivered to the distribution substation, the voltage is first stepped down so that the energy then can be supplied to the smaller industrial and commercial consumers. In contrast, residential consumers normally receive single-phase power from pole-mounted distribution transformers at lower voltages.

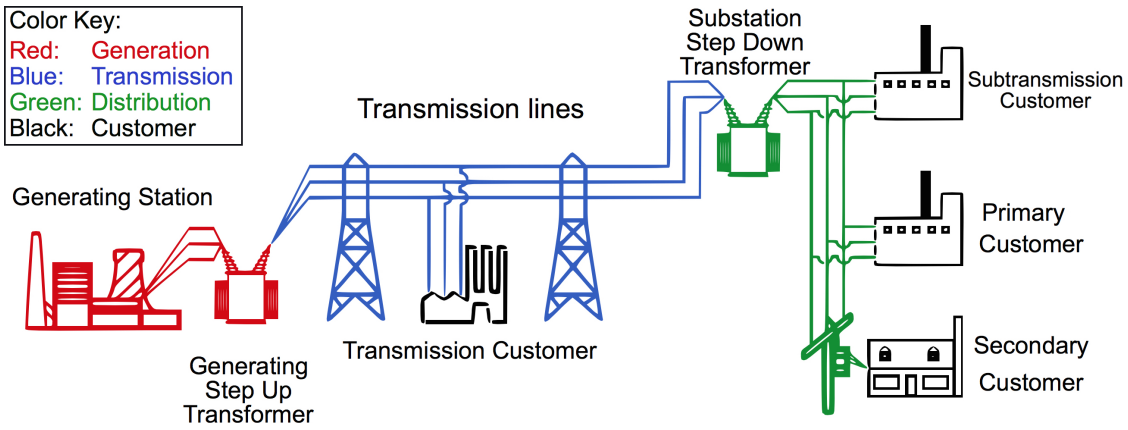


Figure 1.1: Diagram of an electric power system [6]

A conventional electric power system is typically divided into High Voltage (HV), Medium Voltage (MV) and Low Voltage (LV) sections [7], with the interconnections between the different sections called buses, as shown in Figure 1.2. The generator voltages operate in the range of 11kV to 30kV, because it is difficult to achieve higher generator voltage, due to the constrains on the insulation in the narrow confines of the generator stator [5]. However, in order to achieve transmission efficiency, a high transmission-line voltage is required, so transformers are used to step up the voltage to 132kV before the electricity is delivered to the power transmission system. From the power transmission system, electricity is delivered from the generators to the power distribution substation by overhead lines and underground cables. In the power distribution system, the voltage is stepped down to 33kV first, and then later to 11kV. The power distribution system delivers electricity power to primary users and also to the LV network. In the LV network, the voltage is stepped down again to 400V for three-phase and 230V for single phase [8].

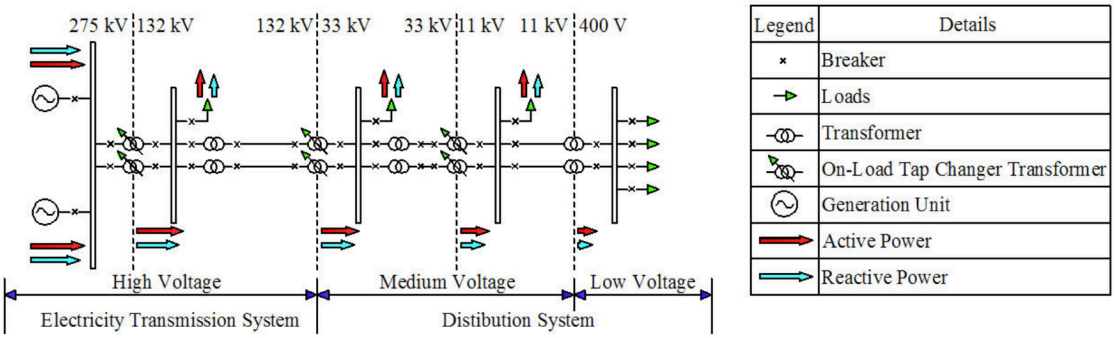


Figure 1.2: Structure of a conventional electric power system [7]

An important property of the conventional power grid is that it cannot adjust itself to react to changes in the supply and demand [9, 10]. The conventional grid has remained unchanged for about 100 years and while the power system is ageing, the demand for electricity power has increased, so the increasing demand for power cannot be handled by the current conventional grid. In order to make the system capable of meeting such a demand, it needs to be upgraded so that it can become more flexible.

1.2 Variability of the supply and demand of power system

1.2.1 The balance in power system

In general, the power generated in the system cannot be stored but is consumed immediately by the loads, as the power system just has limited storage such as batteries and pumped hydro [11–14]. Therefore, there exists a balance between the generated power and the power required by the loads. This balance needs to be satisfied at all times in order to promise the secure operation of the power system. The balance between the supply and demand is not a strict balance but is instead a dynamic balance meaning that the supply and demand are both fluctuating within an acceptable range.

In reality, it is difficult to ensure power generation and power consumption are exactly equal. The nationwide imbalance between supply and demand can be reflected by the fluctuation of the frequency. When the electricity power generated

at the supply side exceeds the power required by the demand side, the frequency will increase, and vice versa. The power system frequency is fluctuating around a fixed nominal point, which in the UK is 50Hz. When the generated power is larger than that consumed by the loads, the frequency of the system will increase. Conversely, when the generated power is smaller than the power demand, the frequency will decrease. In order to keep the balance and prevent accidents, the fluctuation of frequency is normally constrained to 1%, which is 50 ± 0.5Hz [15].

The curve in Figure 1.3 plots the frequency variation of the UK power system between 22:40 to 23:40 on the 8th of May 2018. The data is obtained from the UK nation grid [1]. The vertical axis is the time and the horizontal axis is the frequency, the data is sampled every minute. The signal shows a frequency response constantly changing around the nominal point of 50Hz, but the fluctuation remains within the acceptable range and for most of the time, the frequency is changing smoothly, although sometimes there can be abrupt changes (e.g. at the time point around 23:10, there is an abrupt change in the frequency for about 0.1Hz in 1min).

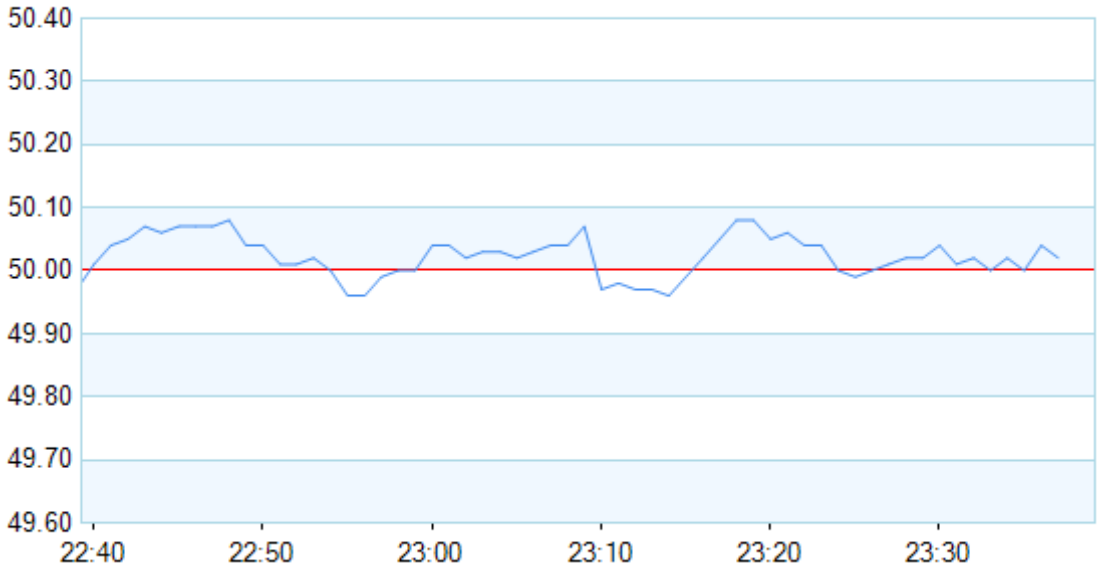


Figure 1.3: Frequency variation over an hour on the 8th of May 2018

[1]The data is provided by the UK national grid website: https://extranet.nationalgrid.com/RealTime

Figure 1.4 shows the frequency variation over a 24h period on the 12th of March 2014, where the data is sampled every 10mins. The data was obtained from Scottish and Southern Energy (SSE) plc., Reading [2]. In the figure, the frequency is constantly changing but again is within the acceptable range. It is also important to have limited fluctuation in the frequency, as strong fluctuation can lead to network failure and damage to the loads.

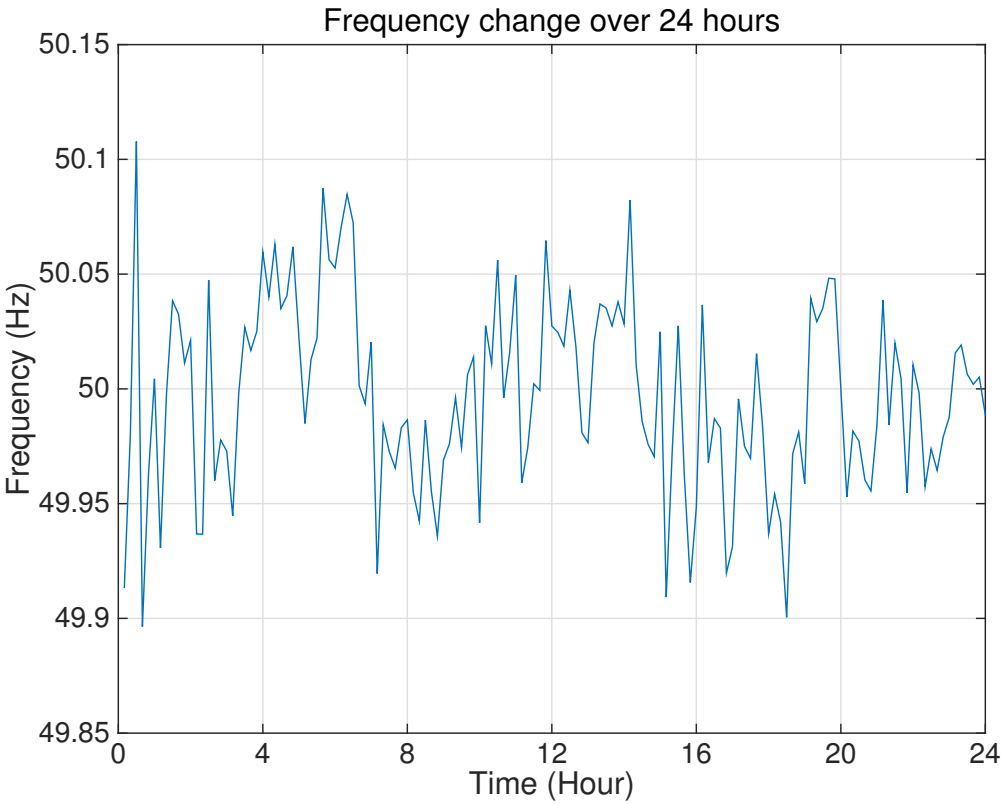


Figure 1.4: Frequency variation over 24 hours on the 12th of March 2014

1.2.2 Variation at supply and demand side

The flexibility of the supply side is normally very limited. When electrical power typically is generated from fossil fuels, regularly turning these generators off and on is known to be expensive, and so accurate demand side predictions are needed [16]. Motivated by the need for energy security and environmental protection, and

[2]The data is provided by Prof. Malcolm D. McCulloch and Dr. Ren Kang from the energy and power group of the Department of Engineering Science, University of Oxford.

to compensate for the rise in fossil fuel prices, the use of variable renewable power electricity generation, such as wind and solar energy, has increased dramatically over the last decades. This has led to a significant increase in the variability of the supply side [17, 18]. Variable generation, such as wind, solar, hydro and tidal, can be defined as those resources whose output is dependent upon the prevailing environmental conditions [19]. An important issue for these variable generation is that the production and scale of the existing power system can lead to difficulties in introducing them, since it is difficult to match these unpredictable generation with demand, particularly for isolated power systems containing mostly generation units with long start up times and low ramp rates [20].

At the same time, there is the increasing use of energy storage technologies, which helps to improve the flexibility of the power supply. As the reliability of renewable power is not as consistent as fuel-based generation, it has been proposed to integrate energy storage devices into the power system network to allow the powers to be buffered [21]. The purpose of introducing these devices into the power network is to store energy when the supply is larger than the demand and the electricity power stored will be used to supply devices in peak time to compensate for the difference between generation and consumption. In the past, only a limited number of such energy storage devices have been incorporated into the power system, primarily due to the lack of availability of tools for operational cost optimisation and the technical issues with large scale storage. It has also been found to be difficult to assess the benefits of storage technologies as batteries are still relatively expensive. However, nowadays with the development of renewable resources such as wind power, which are not controllable and are distributed, it is predicted that energy storage technologies will play an increasingly important role in improving the overall stability and reliability of the grid to improve the power transmission and distribution capacity so as to meet the ever growing power demand [22].

In terms of the demand side, from the conventional perspective, the demand of a power system cannot be controlled or regulated. The supply side needs to have good understanding of the demand and to match the demand with tight tolerances

[23, 24]. A demand curve over a typical day is plotted in Figure 1.5 [25]. From midnight to 4am in the morning, the demand is low as the majority of people are sleeping. Between 4am and 8am, there is a significant increase in power demand, this transition process is called the "morning ramp" [26]. The abrupt variation in demand during this period requires the power generation units to start up quickly. These units can be more costly to operate compared to those that operate over longer periods. There are typically two peak demand periods for electricity during a day, which are around 11am and 6pm. These peaks occur during working hours or after work, as this is when people typically turn on machines like motors, air conditioners, washing machines, etc [27]. The generator units and power transmission and distribution system should be able to satisfy the demand even during peak times. The ability of the power system to supply the demand is called the capacity.

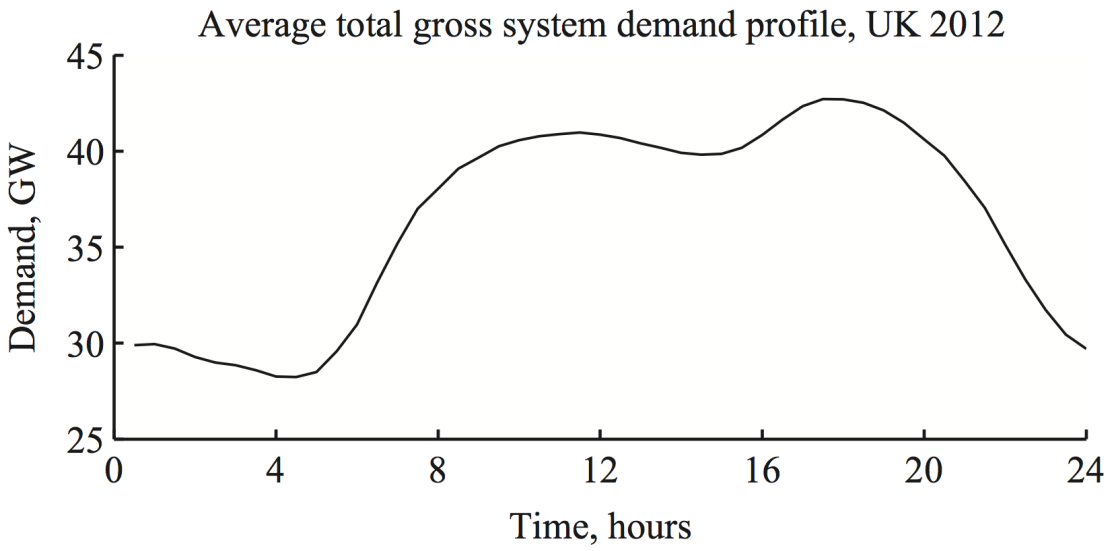


Figure 1.5: Average total power system demand [25]

1.2.3 The concept of demand side management

As mentioned before, from the conventional perspective, the demand of a power system is considered as fixed and there is limited opportunity for controlling or regulating demand. However, in order to improve network efficiency it has been proposed to monitor and regulate the load on-line, thus shaping the demand curve.

This process is called demand side management (DSM), which is the planning, implementation and monitoring of utility activities that are designed to influence the activity of customers and change their electricity consumption patterns. Usually, the main objective of DSM is to encourage users to consume less power during peak times and to shift energy use to off-peak hours so as to flatten the demand curve. Sometimes, instead of flattening the curve, it is more important to encourage the load to change according to the variation of the generation [28].

The operation of the power system requires the balance between load and generation. However, if the load is totally unregulated, it will be very difficult to maintain this balance, which will become even harder when distributed renewable generation is integrated into the system. With the increasing use of renewable power, the power system is facing new challenges [29, 30]. The generation of renewable power depends on the weather and environment and it is generally not easy to modulate the output of renewable power in order to follow the load curve. In addition, peaks in renewable energy do not necessarily coincide with peaks in demand, so energy needs to be stored or artificially consumed by adding additional load into the power network [10]. To cope with these problems and maintaining the balance, the most feasible solution is to engage the activities of the users and control the demand of electricity power.

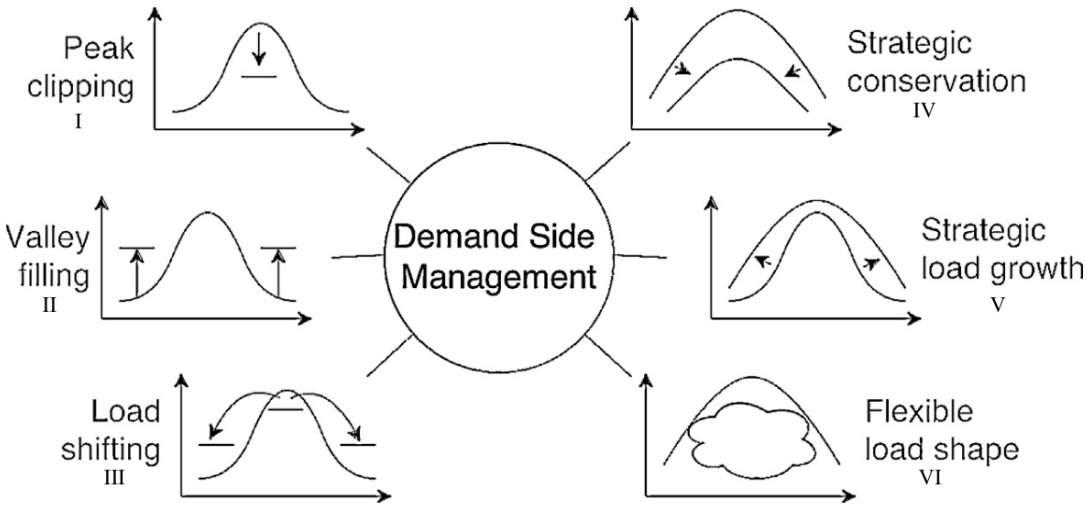


Figure 1.6: Load shaping techniques of DSM [31]

There are many ways to modulate the curve of load, as described in Figure 1.6 [31]. Normally, there are three actions a customer can take in response to change in demand. Customers can reduce load only during critical peak time and maintain normal pattern load during off-peak time as shown by I. This can help the power system to clip the peak demand and thus improve the efficiency, however this will leads to decrease in the overall consumption of the customers, which may make the customers feel dissatisfied. Another approach is to change the time of using loads. For example, people can stop using their washing machine at peak time and turn on these machines again during off-peak hours as shown by III. This can help to shift and flatten the load curve without necessarily decreasing the total electricity consumption [32]. Finally, the customers can also use distributed on site generation to reduce demand seen by the utility as shown by IV and V. This would increase user autonomy, further decentralise generation and decrease the average load on the distribution and transmission grids, although this will increase the complexity of the power system [33]. In addition to these actions, there are also some other demand side management strategies such as valley filling which could flatten the demand curve by adding additional load or power storage device during valley period and flexible load shape which could achieve by introducing flexible loads into the power system to change the demand curve [34, 35].

There are many ways to implement DSM in a power system. Depending on their timing and impact on the customer, they can be categorized into the following four types [22]:

1. Direct Load Control (DLC)
2. Time of Use (TOU)
3. Demand Response (DR)
4. Virtual Power Plant (VPP)

DLC means that the demand is directly controlled. For example, during peak times, machines such as washing machines and air conditioners may be automatically

turned off, before being allowed to be turned back on again after the peak time has passed [27, 36]. By doing this, demand can be shifted in a direct way.

Time of use tariffs (ToU) is a time-varying tariffing approach to achieve DSM. In ToU modalities, a set of different tariffs according to the hours and seasons is set at the beginning of a tariff cycle and kept constant during the whole period [37]. There has been a significant amount of research on optimising the ToU tariff in order to manipulate the demand over time [38, 39]. These results show that ToU can be an effective way to switch the demand to off-peak time. However, the disadvantage of ToU is its lack of flexibility, since the price is set in advance and has to be held for quite a long time.

Demand Response (DR) asks the users to switch their use of electricity power. This requires their voluntary activities [22]. Unlike DLC, which takes the different loads under direct control, DR will broadcast the information indicating the load condition, such as price, to motivate the users to shift their use of energy. The difference between ToU and DR is that in ToU, the information (such as price) is given in advance and fixed for a long time, whereas in DR, there is on-line broadcasting tells users the most up to date load condition information.

Virtual Power Plants (VPPs) are a group of smaller renewable generation units, that appear as a single power plant to the grid management. However, as many renewable resources cannot provide a stable electricity supply (such as wind-driven generator), the implementation of VPP requires complex optimization and distributedcontrol to meet the requirement that the grid should provide predictable and stable power [40].

1.3 The control layers of power system

The power system is a large and sophisticated system which contains different control layers [5, 41]. In terms of the operational objectives and time-scale, these control layers can be divided into mainly two types. There is fast time-scale control applied to the grid such as automatic generation control, which aims at adjusting the power output of the generators according to the change of loads [42, 43] and

turbine governor control which regulates the frequency of the generator and thus influencing the operation frequency of the power system [44, 45]. There is also slow time-scale control applied to the power market such as investments contracts, which provide an internally consistent framework for assigning long-term capacity rights to a sophisticated power transmission network [46, 47] and economic dispatch which determines the optimal output of the generation facilities to meet the system load at the possible lowest cost subject to the transmission constrains [48–50]. The time scale of each layer can be significantly different from the rest and normally when analysing one of the layers, it is commonly assumed that the influence of other layers can be ignored. For example, when designing long-term contracts between a generation company and a factory, the operation of the transmission network, which operates at a faster time-scale, can be neglected. This assumption helps to simplify the analysis of power system.

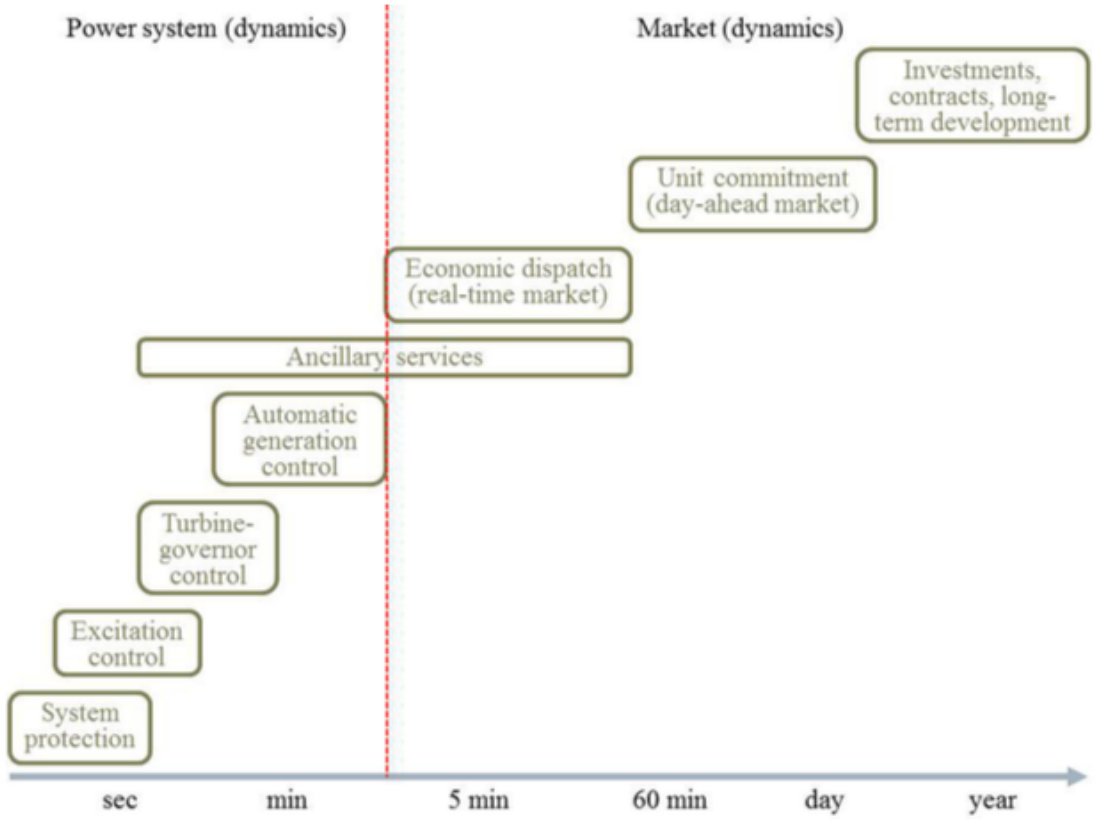


Figure 1.7: Time-scale decomposition of controls and actions in the power grid [51]

In terms of the voltage control in power system, it can also be divided into

different layers according to the object. There are primary layer voltage control relating to excitation control and voltage regulators at the generating stations, which adjusts the voltage from the primary level and operates at at fast time scale (the time level of second) [52–54]. In the secondary layer, which is the power transmission process, the voltage is controlled using on-load tap changer (OLTC) transformers at the transmission end and receiving end, which normally operate several times during a day and ensure that the voltage at the load side can be comparatively stable [55–57]. At the power distribution network and load side which are the tertiary layer, the voltage is adjusted by off-load tap changer, which is operated manually at the frequency of a couple of times a year [58].

In this thesis, we focus on the study of the OLTC transformer which acts an important role in the secondary level voltage control.

1.4 OLTC transformer and tap change

Voltage regulation is a critical aspect of power system operation. As described in Figure 1.2, after electricity power has been generated, it is then delivered to the demand side via the power transmission system using a high voltage to decrease the voltage loss during the power transmission process. At the power distribution station, the voltage is reduced by the transformer. The subtransmission customer and primary customer will then take electricity power from the higher voltage directly. The voltage can be turned down to an even lower level to be used by secondary customers. Thus, the power system operates at multiple voltage levels, in order to provide the stability of the system and security of the users. The distribution network operators (DNO) regulate the voltage supplied to a customer such that it remains within statutory limits. A critical device for secondary level power network voltage control is the on load tap changer (OLTC) in the medium voltage level [59–61]. Between the multiple voltage levels, on-load tap changer (OLTC) transformers are used to maintain the voltage.

A typical OLTC transformer can be viewed as a device with two windings, one for the input and one for the output. On its output winding, the OLTC has

a finite number of winding contacts connecting different winding segments. By connecting the output to different segments, the length of the output winding can be changed, causing the output voltage to change. The change of the transformer ratio made by an OLTC is referred to as a tap change and it is this mechanism that regulates the load bus voltage. As the usage of electricity power changes, the current over the power transmission and distribution cables also changes, and this induces a voltage loss along the cables. In order to balance this voltage loss and keep the voltage at the user side stable, the OLTC transformer makes tap changes that moves its tap to a different winding contact, adjusting its transformer ratio and changing its output voltage [55, 59, 62, 63].

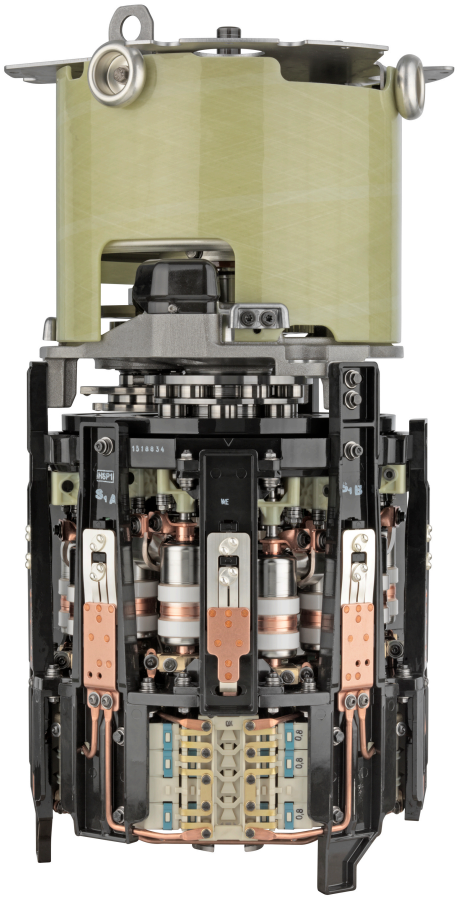


Figure 1.8: An OLTC transformer

1.4.1 OLTC transformer structure

As shown in Figure 1.2, the OLTC transformer is generally used in the power distribution system to transform the voltage from 33kV to 11kV or 6.6kV. A picture of a standard OLTC transformer is shown in Figure 1.8. The OLTC transformer is controlled automatically by a relay that increases or decreases its voltage by moving its tap to different contacts, as shown in Figure 1.9. The contacts are connected to different winding positions. By moving the tap, the OLTC transformer changes the length of its winding and thus changes its transformer ratio and output voltage. Normally, when the secondary voltage detected by an OLTC transformer is no longer within the acceptable deadband, a relay will send a signal to the tap changer mechanism to switch its tap position in order to increase or decrease the voltage so that the secondary voltage can be returned back to the permitted band again. The OLTC transformer along with its voltage control relay, regulates and maintains the voltage magnitude within the required limits.

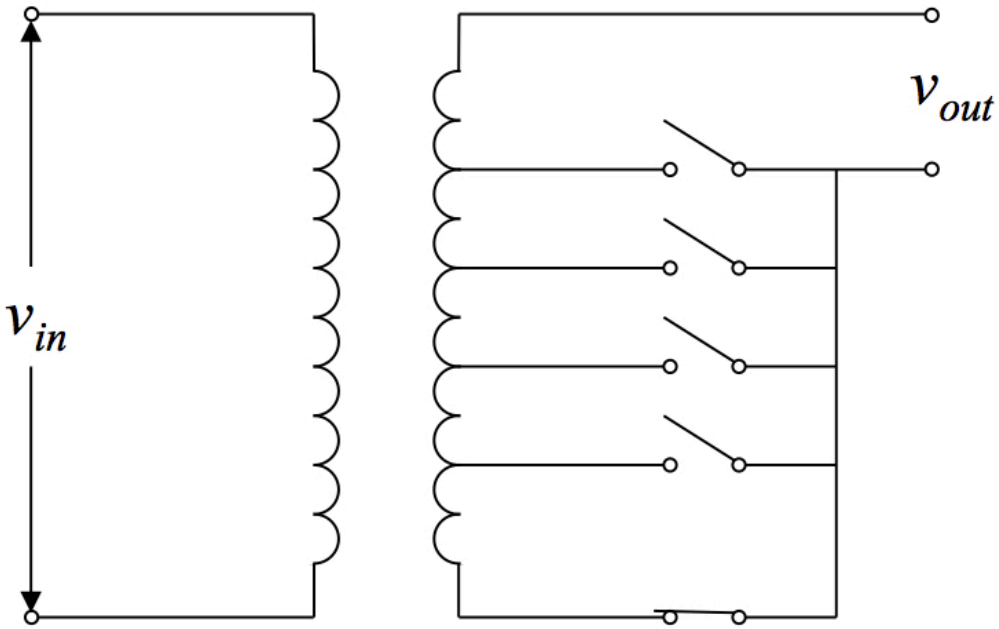


Figure 1.9: The diagram of an OLTC transformer

The movement of a tap changer can be used as an indication of the local load status. The voltage drop over the power transmission and distribution cable

can be expressed as

$$V_d = R_c I, \tag{1.1}$$

where V_d is the voltage drop over the power transmission cable, R_c is the resistance of the cable and I is the current. When the total power consumption in a power distribution network increases, the current I also increases which causes the voltage drop to increase as well. This leads to a voltage drop at the load side and finally to a step up tap change. In contrast, when the power consumption decreases, a step down tap change will be made. So, when the OLTC makes a step up tap change, it indicates that the load is increasing and when a step down tap change is made, it implies that the total load is decreasing.

1.4.2 Tap changes

The typical step size of an OLTC transformer is 1.25% or 1.43%. In power system, a tap change will affect the load bus voltage, which can be detected by the users. The voltage variation of a typical tap change is plotted in Figure 1.10 , and Figure 1.11 shows a section of the detail of Figure 1.10, which is highlighted by the red rectangular. The real voltage data is provided by Professor Malcolm D. McCulloch and Ren Kang from the Energy and Power Group of the Department of Engineering Science, University of Oxford, which was obtained from Scottish and Southern Energy (SSE) plc., Reading.

The figure shows the top section of a set of sinusoidal voltage signal. The sampling frequency is $f_s = 10$kHz, and the voltage frequency is around 50Hz. The top section of the sinusoidal signal shows the outline of the voltage magnitude variation. It can be seen that before a tap change happens, the voltage is around 331.5V. At $t = 125.6$s, a tap change is triggered and the voltage jumps to about 334V in less than 0.4 second. In a real power distribution network, tap changes like this will happen 4 to 6 times a day, including both step up and step down tap changes.

Recently, renewable energy sources have been increasingly integrated into the distribution network. This will affect the local balance between supply and

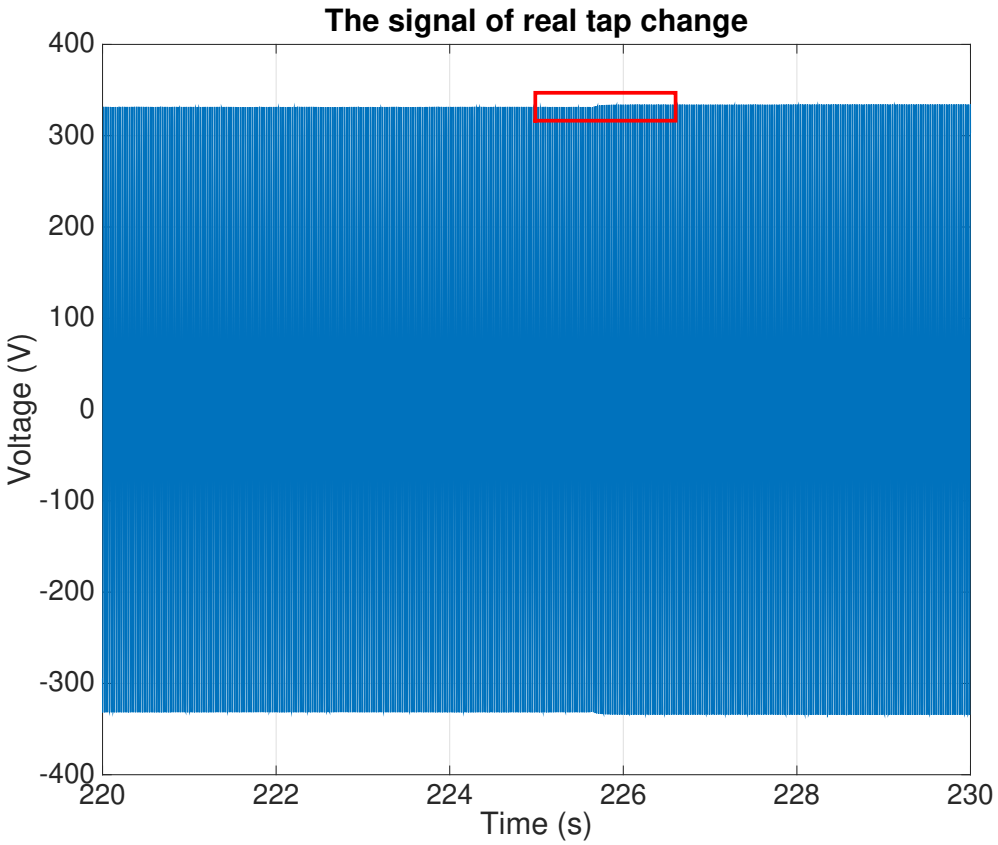


Figure 1.10: Signal of a real tap change

demand and therefore influence the power flow and the voltage regulation. The integration of renewable energy resources will result in an voltage increase at the point of connection, which will affect the movement of tap changers. Thus, OLTC transformer control method needs to be improved as it needs to be more flexible and smarter [64–66].

1.5 Outline of thesis

This thesis focus on the detection and analysis of tap changes and it contains five chapters. Following this introduction chapter, the second chapter, considers the problem of detecting the frequency of power system. As a very important parameter of power system, voltage frequency plays an essential role in the detection of tap changes and it can also be used for indicating the imbalance between supply and

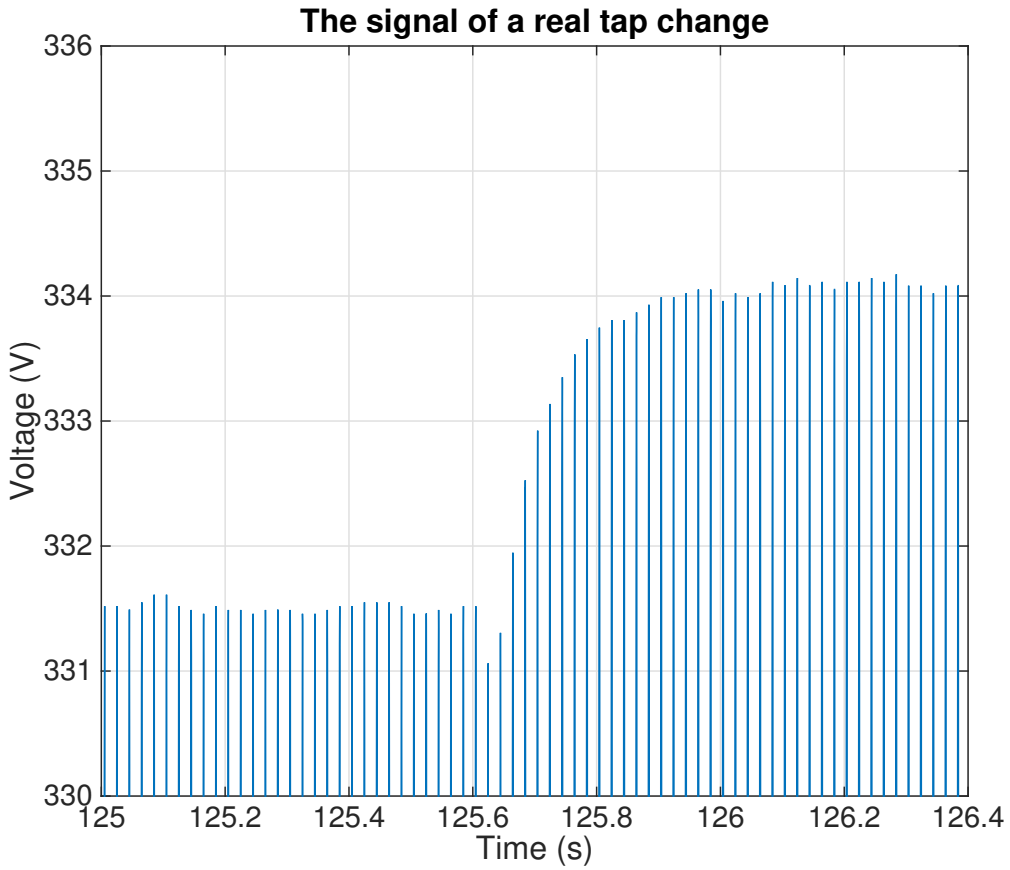


Figure 1.11: Signal of a real tap change (detail of the top portion highlighted by the red rectangular in Figure 1.10)

demand in power system. Considering the limit of the current frequency estimating methods, we proposed a new algorithm that can measure power system frequency accurately and track the frequency variation quickly. In order to achieve better performance, the PLL or the Prism algorithm [67] are used to provide an initial estimate of the frequency, and a complex least mean square algorithm is used to refine the initial frequency estimate so that a more precise frequency result can be obtained. The frequency estimation algorithm is tested with both of simulated data and real data and the performance is compared with other methods.

The third chapter focus on the detection of tap changes. As tap changes can be good indicator of the local network stress, it is very important to detect when a tap change has happened in power system. In this chapter, an algorithm for automatic detection of tap changes on an electricity grid is proposed, which can find

tap change from the voltage fluctuation accurately. In order to detect tap changes in a power distributed network, there are three parts of the detection algorithm. The first part is a phase-locked loop (PLL) that can trace the operating frequency of the voltage signal. The second part is a windowed recursive least square (WRLS) algorithm, which is used to calculate the magnitude of the harmonics within the voltage signal. The third part is a hypothesis test, which is used for detecting if there is a tap change or not. The algorithm is tested with a set of real time data obtained from the real power system.

The fourth chapter focuses on the modelling of a tap changer in power system and proposes a solution for removing the chattering problem caused by a quantiser tap changer model. The currently used tap changer models are either have continuous output or working in discrete time, making them not suitable for control applications. The proposed tap changer model combines a linear dynamic function with a quantiser, which evolves in continuous time but has a quantised number of tap positions. The simulation results of this tap changer model connected with finite and infinite generator bus models are presented and discussed in this chapter. The last chapter concludes the thesis.

1.6 Contributions

The main contributions of this thesis are:

- The development of an algorithm for power system frequency estimation.
- A comparison of frequency estimation methods with different types of input signal.
- The development of an automatic tap change detection algorithm that can detect tap changes on an electricity grid, which can provide the information about the local load stress.
- The development of a continuous time tap changer model that has quantised tap positions.

- A comparison of various combinations of tap changer and generator bus models commonly used in the literature.
- A demonstration of the effect of chattering in the model together with a method to remove it using a deadzone.

2

Frequency estimation of power system

Contents

2.1 Introduction **21**
2.2 Structure of the algorithm **23**
2.3 Initial estimation method **24**
2.3.1 Phase-locked loop 24
2.3.2 Prism algorithm 33
2.4 CLMS algorithm for result enhancement **35**
2.4.1 Fixed step CLMS enhancement method 35
2.4.2 Adaptive step CLMS enhancement method 38
2.5 The combined frequency estimation algorithm **40**
2.5.1 Fixed step CLMS enhancement method 40
2.5.2 Adaptive step CLMS enhancement method 42
2.6 Simulation and Discussion **44**
2.6.1 Fixed step CLMS enhanced HTPLL algorithm 44
2.6.2 Fixed step CLMS enhanced Prism algorithm 50
2.6.3 Adaptive step CLMS enhanced HTPLL algorithm . . . 56
2.6.4 Adaptive step CLMS enhanced Prism algorithm 63
2.7 Conclusion **69**

2.1 Introduction

In power systems, one of the most important parameters is the system frequency. Under steady-state conditions, the total power generation should be equal to the sum of power consumption. In such circumstances, the frequency will stay at its

nominal value, which in the UK is 50Hz [8]. However, in real power systems, it is difficult to guarantee the strict balance between generation and consumption. Under such conditions, the voltage will drift from its nominal value, so that when the total generation is larger than the power required by the system, the frequency will increase and be higher than the nominal value, and vice versa [68]. The operation of many loads on the power system requires a relatively stable voltage frequency and in order to ensure that these loads can operate properly and to maintain the stability of the system, there is a requirement for the fluctuation range of the frequency, which is 50Hz±1% in the UK [69]. The variation of frequency reflects the stability of power system, so it is important to know the operating frequency [8, 70]. In addition, when measuring other parameters of the power system, such as the total harmonic distortion (THD), the precise value of frequency is also required [71].

If the voltage signal in a power system is considered as a pure sinusoidal wave, the time between two zero crossing points can be used as an indication of the system frequency [72, 73]. However in reality, the voltage signal is a distorted wave containing high harmonics and noise, so detecting the zero crossing time is not an ideal approach to estimating frequency. Many researches have developed frequency tracking techniques for the voltage signal in power system, but these methods have shortcomings. The discrete Fourier transform (DFT) method is very popular and easy to implement, but will lead to a trade-off between its time resolution and frequency resolution, and the frequency leakage problem cannot be ignored [74, 75]. Phase-locked loops (PLLs) can track the frequency accurately, but in order to achieve good tracking performance, the phase detector needs to be picked carefully and the loop filter should be well-designed [76–78]. Approaches based on an extended Kalman filter can also be used for tracking the voltage frequency, but when the high harmonic contents are included, the computational cost is significant [79–82]. The methods using complex least mean square (CLMS) technique are sensitive to noise [83–87].

In this chapter, we will focus on the estimation of power system frequency and propose an enhanced frequency estimation algorithm which can track the voltage

frequency quickly and accurately. The proposed algorithm takes the three-phase voltage data as the input signal, and is based on the combination of Phase-locked loops (PLL) or Prism algorithm and the CLMS method. In order to achieve better frequency tracking performance, the PLL used in the algorithm is a well-designed HTPLL, which is used to provide an initial estimate of the frequency. The Prism algorithm is also used and tested as the initial frequency estimation technique. An $\alpha\beta$-transform is used to form a complex voltage vector, which is taken as the input of the CLMS algorithm [5, 88–90]. Combining the initial estimate and the complex voltage vector, the CLMS algorithm can refine the estimate to provide a final frequency estimate. The performance of the algorithm is evaluated using both simulated inputs and data from a power system. The results are compared to that of the HTPLL and CLMS on their own, showing the benefit of the combined CLMS enhanced algorithm.

2.2 Structure of the algorithm

A diagram of the complex least mean square (CLMS) enhanced Hilbert-transform phase-locked loop (HTPLL) approach for frequency estimation is shown in Figure 2.1. The algorithm for CLMS enhanced Prism algorithm has similar diagram. This algorithm is designed to estimate the frequency of the three-phase voltage signal in the power system, where V_a, V_b and V_c are the three-phase signals in power system, and between any two of them, there is a $\frac{2\pi}{3}$ phase difference. It is worth mentioning that in a real power system, the voltage is not a pure sinusoidal wave, but a distorted signal containing harmonics, which is caused by effects including saturation of the upstream transformers, the downstream loads and the unexpected disturbances in the power transmission process. In order to obtain a clear signal and achieve a good frequency estimation, the voltage needs to be processed by a pre-filter. The pre-filter can be a low-pass filter or a notch-filter which filters the high harmonics. In this algorithm, the pre-filter is a notch-filter achieved using Prism method [67, 91].

After being processed by the pre-filter, the three-phase signals are put into three copies of a well-designed HTPLL and the mean value of the outputs from

the HTPLLs are taken as an approximate estimate of the frequency. Even though pre-filters are used to eliminate the high harmonics, due to the noise signal in the input and the transient performance of HTPLL, when the input frequency changes, the HTPLL may not be able to provide a precise estimate of the frequency. Therefore the approximate estimate is put into the CLMS algorithm.

As shown in the figure, in order to obtain an expression of the complex voltage that can be processed by the CLMS algorithm, the $\alpha\beta$-transform is used [79]. It transforms the three phase input V_a, V_b and V_c into a complex vector, which contains the phase information, and this complex vector is input to the CLMS algorithm. Combining the complex vector obtained and the approximate estimate from the HTPLL, the CLMS algorithm provides a refined estimate of the voltage frequency.

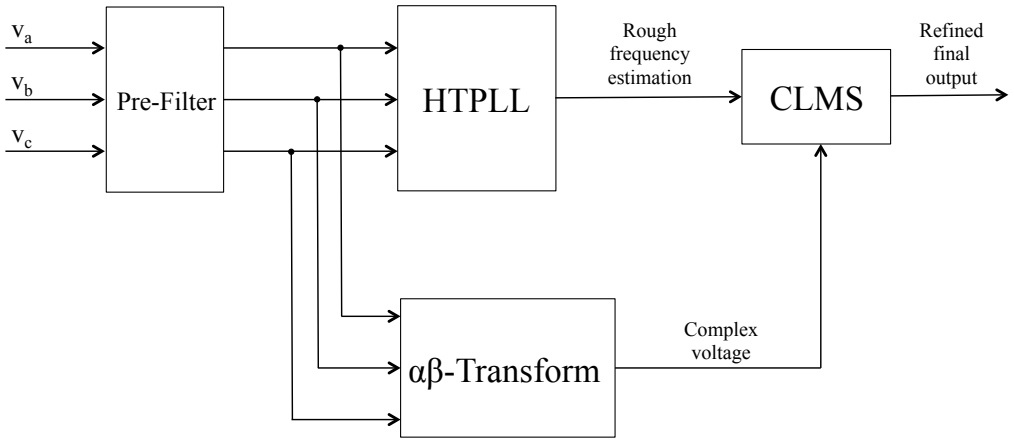


Figure 2.1: The diagram of complex LMS enhanced HTPLL algorithm for frequency estimation in three-phase power system

2.3 Initial estimation method

In this section, the initial estimation methods used for the algorithm are introduced, including the Hilbert-transform phase-locked loop and the Prism pre-filter.

2.3.1 Phase-locked loop

There are many different techniques that can be used as the initial frequency estimation method for this algorithm. Considering the tracking accuracy, one of the method we adopted is a PLL.

Diagram of PLLs

A PLL is a circuit that can generate an output signal with the same frequency as an input or reference signal. Even though there are many different kinds of PLLs, they generally consist of three parts, which are Phase Detector (PD), Loop Filter (LF) and Voltage-Controlled Oscillator (VCO) [92–94], as illustrated in Figure 2.2.

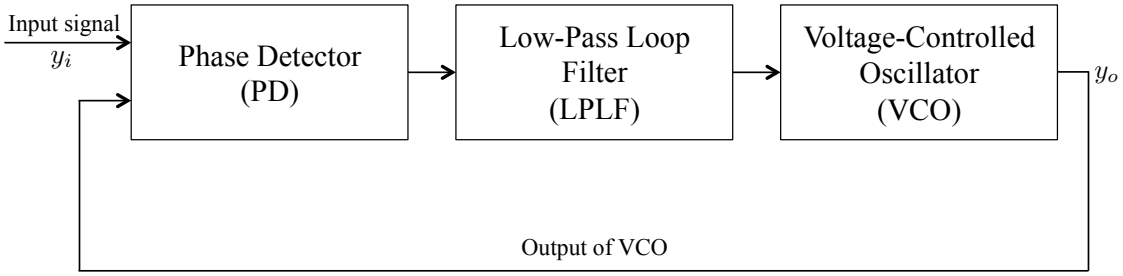


Figure 2.2: Diagram of PLL. The phase detector, low-pass loop filter and voltage-controlled oscillator are connected together to form a PLL, which is a close-loop system.

The phase detector (PD) is a circuit that detects the phase difference between input and reference signal. The output of the PD is designed to be proportional to that phase difference, so that this signal can be used to control the output frequency of the PLL.

In order to improve the signal-to-noise ratio and improve the PLL's ability of tracking dynamic varying input frequencies, a low-pass loop filter is needed and usually, first or second order low-pass filters are used. A third order low-pass filter is rarely used since it can make the system unstable if not properly designed [95].

The Voltage-controlled oscillator (VCO) is a circuit that can generate a variable frequency output signal, where the frequency is controlled by its input voltage. For a PLL system, the output of the VCO is fed back to the PD, so that the PLL can adjust its output until the output frequency is equal to the input frequency.

Classical PLLs

A classical PLL contains all three main components of PLL, namely PD, LF and VCO, and uses a multiplier as the PD. The input signal of a PLL can be expressed as $y_i = A_i \sin(2\pi f_i t + \theta_i)$, where the notation i signifies input. The output of

the VCO is $y_o = A_o \cos(2\pi\, f_o\, t + \theta_o)$, where the notation o means output. In our algorithm, the input signal y_i is the periodic signal obtained from power distribution system. Supposing that the input and output of a classical PLL are both pure sine wave signals and ignoring the noise and high harmonic contents, then for a multiplier-type PD, which compares the phase different between the input and output by multiplying them directly, we have

$$y_i \cdot y_o = A_i \sin(2\pi\, f_i\, t + \theta_i) \cdot A_o \cos(2\pi\, f_o\, t + \theta_o), \tag{2.1}$$

which can be rearranged as

$$y_i \cdot y_o = \frac{1}{2} A_i\, A_o\, [\sin(2\pi\, (f_i + f_o)\, t + \theta_i + \theta_o) + \sin(2\pi\, (f_i - f_o)\, t + \theta_i - \theta_o)]. \tag{2.2}$$

From (2.2), we can see that the output of the classical PD contains a high frequency component and a low frequency component, they are

$$C_h = \frac{1}{2} A_i\, A_o \sin(2\pi\, (f_i + f_o)\, t + \theta_i + \theta_o), \tag{2.3}$$

$$C_l = \frac{1}{2} A_i\, A_o \sin(2\pi\, (f_i - f_o)\, t + \theta_i - \theta_o). \tag{2.4}$$

When the PLL is working in steady state, its output signal should have the same frequency as the input signal, which is called lock-in status. When a classical PLL is locked in, its output frequency f_o equals to the input signal frequency f_i, that is $f_o = f_i$. Then (2.2) can be rewritten as

$$y_i \cdot y_o = \frac{1}{2} A_i\, A_o\, [\sin(4\pi\, f_i\, t + \theta_i + \theta_o) + \sin(\theta_i - \theta_o)]. \tag{2.5}$$

From (2.5), when a PLL is locked in, the high frequency component of its PD output is twice the frequency of the reference signal, while the low frequency content becomes a constant. Although the constant part $\sin(\theta_i - \theta_o)$ is not linear to the phase error, when phase error $\Delta\phi = \theta_i - \theta_o$ is small, the sine function $y = \sin\Delta\phi$ can be approximated by a linear function $y \approx \Delta\phi$. In terms of the voltage controlled oscillator, it generates an output sinusoidal signal and the frequency of the output is proportional to the VCO's input. The phase of the output sinusoidal signal is the integration of the input voltage, so the VCO can be viewed as an integrator.

With the analysis above, the classical PLL system can then be approximated by a linear system. For the high frequency component, which is at the frequency of the second harmonic, a low-pass Loop Filter (LPLF) is used to remove this component, leaving only the constant component.

The design of the LPLF is very important for the design of PLL, as the performance of the LPLF will affect the PLL's responses considerably. Higher order loop filters can filter the high harmonic contents better but are more likely to result in unstable issues, whereas lower order loop filters are easier to design but will leave more high harmonic contents in the filtered signal and normally their dynamic performance are not as good as those of the higher order loop filters. In this research, the PLL is used to track the frequency of the power system. In a real power system, the frequency variation is shown in Figure 1.3, and even though the voltage frequency was constantly changing and sometimes could change abruptly, these variations happened in a larger time scale (the data in the figure is sampled every minute). It is worth to be mentioning that for the algorithm proposed in this research, the input signal sampling frequency is 10kHz, which means the sampling time is considerably shorter than the data used for the figure. It is reasonable to assume that in a comparatively short period of time, the frequency is changing following a slope curve. To track a slope changing input with zero steady-state error, the system should be a Type 2 system or higher. In order to simplify the design process, we adopt a Type 2 system PLL for the algorithm, which leads to a first order LPLF. The simplest first order LPLF is a PI controller [96], which has transfer function

$$C(s) = K\left(1 + \frac{1}{s\,T_i}\right). \tag{2.6}$$

Combining the three components of the PLL, the closed-loop transfer function is

$$G_e(s) = \frac{G_o(s)}{1 + G_o(s)} = \frac{\dfrac{KA}{2T_i} + s}{s^2 + \dfrac{KA}{2}s + \dfrac{KA}{2T_i}}, \tag{2.7}$$

where $A = \frac{1}{2} A_i A_o$. Comparing this with the standard second order transfer function, we can see that the natural frequency for this transfer function is

$$\omega_n = \sqrt{\frac{KA}{2T_i}}, \tag{2.8}$$

with damping factor

$$\zeta = \sqrt{\frac{KAT_i}{8}}. \tag{2.9}$$

In this system, ω_n and ζ are design parameters that are determined by the parameter K and T_i for the LPLF. The natural frequency ω_n will affect the PLL's reaction speed and also influence the amplitude of the high harmonic content in the output of PD. In fact, while designing the LPLF, there is a trade-off between the reaction speed of PLL and the efficiency of LPLF: a large ω_n can provide faster reaction and better lock-in speed for the PLL against a step change, but on the other hand, this will also increase the amplitude of the high harmonic component in PD output and increase the overshoot in the transient process of the PLL output as well.

Hilbert-transform PLL

Even though the output of the traditional PLL can reflect the frequency of voltage signal in the power grid, the second harmonic content from its PD introduces an error in the estimated frequency and if we use this PLL as part of the magnitude calculation, this will lead to an error in the estimate of the magnitude. In order to obtain better frequency information and thus achieve better magnitude estimation, we can substitute the classical PLL by a Hilbert-Transform PLL (HTPLL) for frequency tracking [97].

HTPLL is a PLL that uses a Hilbert Transformer Phase Detector (HTPD). The Hilbert transformer is a system that can generate an output signal whose phase is delayed by $\pi/2$ compared to the phase of the input signal. As the VCO can easily generate both $\sin\phi_o$ and $\cos\phi_o$, from the input signal $\cos\phi_i$, we can obtain $\sin\phi_i = \cos(\phi_i - \pi/2)$ using a Hilbert transformer. Then, using inverse trigonometric functions, we can obtain the phase error between the input signal of the PLL and

the output signal of VCO. It worth mentioning that in this process, the phase error is not obtained via multiplying the input and VCO output directly, thus there will not be the second order harmonic component in the output of HTPD.

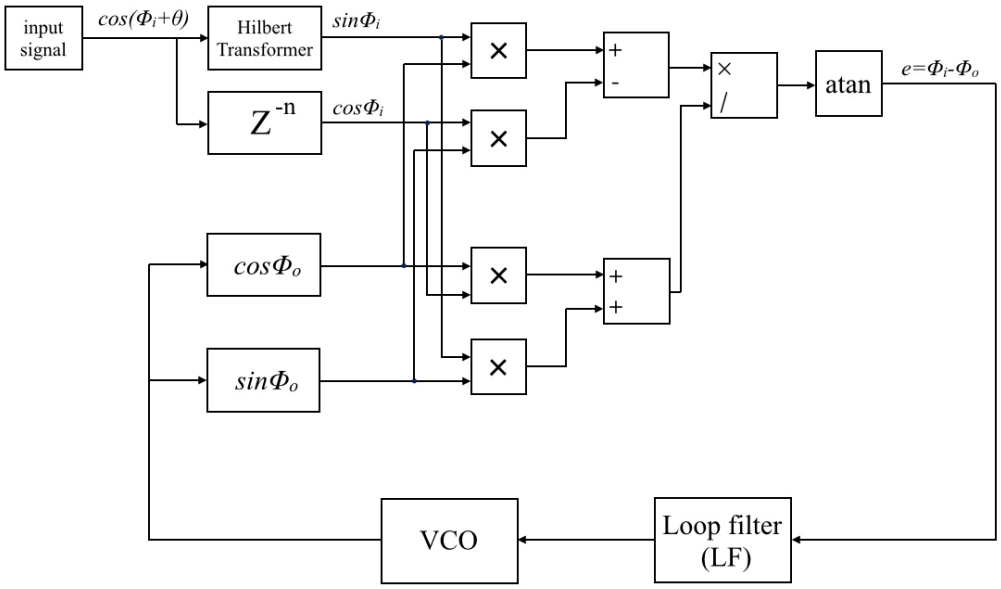


Figure 2.3: Diagram of a HTPLL [96]

Figure 2.3 gives the block diagram of Hilbert transformer phase detector, which shows how the HTPD detects the phase error. It can be seen that, the input signal is applied to both a Hilbert transformer and a delay function, so that we have $A_i \cos(2\pi f_i t + \theta_i)$ and $A_i \sin(2\pi f_i t + \theta_i)$ and from the VCO, we can obtain $A_o \cos(2\pi f_o t + \theta_o)$ and $A_o \sin(2\pi f_o t + \theta_o)$. Let $\phi_e = \phi_o - \phi_i$, then since

$$A \cos \phi_e = A \cos(\phi_o - \phi_i) = A \cos(2\pi f_i t + \theta_i - 2\pi f_o t - \theta_o), \tag{2.10}$$

where $A = A_i A_o$, so that

$$A \cos \phi_e = A_i \cos \phi_i \cdot A_o \cos \phi_o + A_i \sin \phi_i \cdot A_o \sin \phi_o. \tag{2.11}$$

Similarly,

$$A \sin \phi_e = A_i \sin \phi_i \cdot A_o \cos \phi_o - A_i \cos \phi_i \cdot A_o \sin \phi_o, \tag{2.12}$$

and

$$S_e(t) = \tan^{-1} \frac{A \sin \phi_e}{A \cos \phi_e} = (\phi_i - \phi_o) \mathrm{mod}(2\pi). \tag{2.13}$$

where mod(x) indicates modulo. In this way, we can obtain S_e, which is the phase error between the input signal of PLL and the output signal of VCO.

In order to implement the Hilbert transformer as part of a PLL, we need an operation that can shift the phase of the input signal by $-\pi/2$ without affecting its amplitude. This filter should have the (non-causal) pulse response as follows

$$h(n) = \begin{cases} \frac{2}{\pi}\frac{\sin^2(\pi n/2)}{n}, & when\ n = \pm 1, \pm 2, \cdots \\ 0, & when\ n = 0, \end{cases} \tag{2.14}$$

and the frequency response of this filter is

$$H(jw) = \begin{cases} -j, & 0 \leq \omega < \pi, \\ j, & \pi \leq \omega < 0. \end{cases} \tag{2.15}$$

From the response, we see that the phase shift for this filter is always $\pm\pi/2$, and that the length of the pulse response of this filter is infinite. In order to produce a practical filter, we need to limit the length of the pulse response to a finite number m to approximate the infinite number of terms and causality is imposed by processing the input signal both to the filter and a delay amounting, which will delay the signal to half of the length of the finite pules. Then the desired $\pi/2$ phase change can be achieved.

Based on Parks-McClellan algorithm [98], we can obtain a good estimate for Hilbert Transformer, which can be achieved easily with Matlab filter design function *firpm* and filter function *filter*. The order of the filter, passband frequency and the required amplitude can be set in this function. The frequency set in the *firpm* function should be the normalised frequency f', which can be calculated by $f' = f/f_N$, where f_N is the Nyquist frequency, which for sampling frequency f_s is $f_N = f_s/2$. In our problem, the sampling frequency is $f_s = 10^4$Hz and the input signal frequency f_i is around 50Hz, so the normalised lower cut-off frequency of the passband filter should be

$$f_1' = f_i/f_N = 2f_i/f_s = 0.01,$$

and in order to achieve symmetrical normalised passband frequency, the higher cut-off frequency is set to be $f_2' = 0.99$. The filter should have unity gain 1, which means

the output signal should have the same amplitude with input signal. The order is chosen to be 600, which corresponds to three periods of the input sine wave signal at frequency 50Hz. It should be noticed that in order to achieve $\pi/2$ phase delay, a delay function which can delay the input signal by half of the filter order length is also needed, as shown in Figure 2.3. The phase difference between the output signal of the Hilbert transformer and the output of the delay function will be exactly $\pi/2$.

An HTPLL has all of the three main sections of a PLL, so a low pass loop-filter and a VCO are also required. As the only difference between our HTPLL and the classical PLL is the type of phase detector, this will not affect the transfer function of the system. Following the PLL designing method proposed in [96], the HTPD's output is clearer, which can benefit the performance of the PLL. So the PLL picked for the initial frequency estimation section is a HTPLL.

Design of HTPLL

In order to achieve zero steady-state error when tracking slope changes in frequency, when using a PI controller as the LPLF as described in (2.6), since the HTPD has unity gain ($A = 1$), the closed-loop transfer function from the input phase to the output phase of the HTPLL is

$$G_e(s) = \frac{\frac{K}{2T_i} + s}{s^2 + \frac{K}{2}s + \frac{K}{2T_i}}. \tag{2.16}$$

Comparing (2.16) with the standard second order transfer function, the natural frequency of the HTPLL with a PI controller LPLF is

$$\omega_n = \sqrt{\frac{K}{2\,T_i}}, \tag{2.17}$$

with damping factor

$$\zeta = \sqrt{\frac{K\,T_i}{8}}. \tag{2.18}$$

These parameters has the same mathmetical expression as that of the classical PLL. In (2.17) and (2.18), ω_n and ζ are design parameters, and in order to decrease

the overshoot of the system in transient process and achieve good tracking, we choose $\zeta = 1$, so that there is no overshoot in the phase

$$K = \frac{4\,\omega_n}{A}, \tag{2.19}$$

$$T_i = \frac{2}{\omega_n}. \tag{2.20}$$

From (2.19) and (2.20), it can be seen that as long as the natural frequency ω_n is fixed, the parameter K and T_i in HTPLL transfer function (2.16) are fixed. However, the choice of ω_n is arbitrary. Figure 2.4 shows the responses of HTPLLs with different values for ω_n to the same input frequency step change. The input signal frequency is $f = 50$Hz from the start of the simulation. At time $t = 15$s, there is a step change in the input frequency, so that it changes from $f_1 = 50$Hz to $f_2 = 50.2$Hz. The simulated input signal is a pure sinusoidal wave with no noise. Note that although $\zeta = 1$, so that there is no overshoot in the phase response, but there is overshoot in the frequency. The only difference between the HTPLLs is the ω_n value. In order to compare the performance of the PLLs with different ω_n, we defined the transient time to be the period from when the change happens to the time when the output falls around the expected output within 5% range of the amount of change, which here will be $f_2 \pm 5\%(f_2 - f_1) = 50.2 \pm 0.01$Hz. According to (2.19), (2.20) and Figure 2.4, it can be seen that a larger ω_n leads to a smaller T_i and shorter lock-in time, but increases the LPLF gain K, which results in more noise signal in the LPLF output. This coincide with the numerical result shown in Figure 2.4. The transient time for PLL with $\omega_n = 0.5\pi$ is about $T_t = 1.8$s; PLL with $\omega_n = \pi$ has the transient time of $T_t = 0.9$s; when $\omega_n = 2\pi$, the transient time is $T_t = 0.5$s; $\omega_n = 4\pi$ leads to a transient time of $T_t = 0.3$s and $\omega_n = 8\pi$ making the PLL lock-in in about 0.2s. In this case, as the PD used in HTPLL is a HTPD, which does not lead to the second order harmonic in its output and the input signal is a pure sinusoidal signal, the noise issue is not obvious. When processing real data, the input signal will be a distorted wave with noise, the LPLF should have good filtering ability, so ω_n cannot be too large. In order

to achieve good compromise between noise cancelling performance and frequency tracking, we choose $\omega_n = 4\pi$ for the HTPLL.

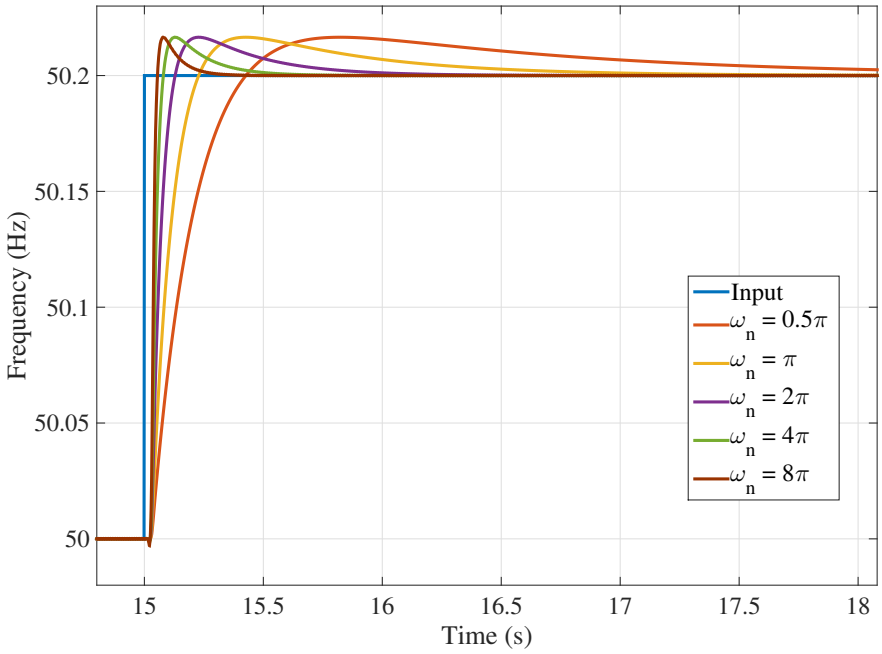


Figure 2.4: Comparison of the performance of HTPLLs with different ω_n

2.3.2 Prism algorithm

The pre-filter is based on the Prism algorithm [67, 91], which is a dual-layered signal processing object containing six networked integration blocks. These blocks accept time series inputs and generate corresponding output time series at the same rate. The integration blocks are multiplying the input with a modulating function with frequency hmHz, where h is the harmonic order of Prism and m is the characteristic frequency. The algorithm characteristic frequency m is a design parameter.

For a sinusoidal input $s(t) = A\sin(2\pi ft + \phi_i)$, with amplitude A, frequency f, and initial phase ϕ_i. The output of the Prism block can be described as the following format,

$$I_{[s|c]_h[s|c]_h} = \int_{-\frac{1}{m}}^{0} [\sin\,|\,\cos](2\pi hmt_1') \left(\int_{t-\frac{1}{m}}^{t} [\sin\,|\,\cos](2\pi hmt_2') \cdot s(t)dt_2' \right) dt_1'. \quad (2.21)$$

The Prism algorithm output, has two outputs, which are $G_s^h = I_{ss}^h + I_{cc}^h$ and $G_c^h = I_{cs}^h - I_{sc}^h$ and by combining these outputs, the input signal frequency f can be calculated [91]. The Prism frequency tracking method used here is the recursive signal tracker (RST). The RST uses one Prism and the past values of G_s and G_c to calculate the frequency of an input sinusoidal in the input. From the Prism algorithm described above, we can obtain two output which is the G_s and G_c

$$G_s = A\ \mathrm{sinc}^2(r)\frac{r^2}{r^2-1}\sin(\phi - 2\pi r), \tag{2.22}$$

and

$$G_s = A\ \mathrm{sinc}^2(r)\frac{r}{r^2-1}\cos(\phi - 2\pi r), \tag{2.23}$$

where frequency of the input relates to r and ϕ, where $r = f/m$ is the frequency ratio, f is the input signal frequency and m is the frequency of the Prism, which is a design parameter.

Given an initial estimate of r, which is $\hat{r}$, then combining $\hat{r}$ with the corresponding Prism output G_{s0} and G_{c0} and the Prism output $1/m$ seconds earlier, which is G_{sm} and G_{cm}, the current frequency can be calculated with the following equation.

$$\hat{\phi_0} = \arctan\left(\frac{G_{s0}}{\hat{r}G_{c0}}\right) + 2\pi\hat{r}, \tag{2.24}$$

$$\hat{\phi_m} = \arctan\left(\frac{G_{sm}}{\hat{r}G_{cm}}\right) + 2\pi\hat{r}. \tag{2.25}$$

Then an improved estimating value of r is given by

$$\hat{r} = (\hat{\phi_0} - \hat{\phi_m})/2\pi. \tag{2.26}$$

The improved frequency estimate is calculated with $\hat{f} = \hat{r}m$.

The Prism RST algorithm response to a step change in input sinusoidal frequency is shown in Figure 2.5. The input signal is a pure sinusoidal signal which is sampled at the sampling frequency of $f_s = 10$kHz. The input frequency changes from $f_1 = 50$Hz to $f_2 = 50.2$Hz at $t = 10$s. The frequency of the Prism is set to be twice of the input frequency, which is $m = 100$Hz and it can be seen from the figure that the Prism algorithm can track the input frequency change quickly. We adopt similar definition of transient time as for the comparison of PLLs, which is

from the start when change happens to the time when the output stays around the expected output within 5% of the change amount (50.2 ± 0.1Hz). After the input frequency change happens at $t = 10$s, the Prism algorithm tracks to the new value in about $T_t = 0.06$s. The yellow curve in the figure plots the Prism algorithm performance with a pre-filter. The pre-filter used here is a Prism notch filter which can filter the noise signal and the high harmonics in the signal, although due to the pre-filter sliding window, there will be a delay in the output of the algorithm. It can be seen from the figure that it can still track the input frequency variation in about $T_t = 0.36$s. So, even when a pre-filter is added, the response time of the Prism algorithm can still be very short.

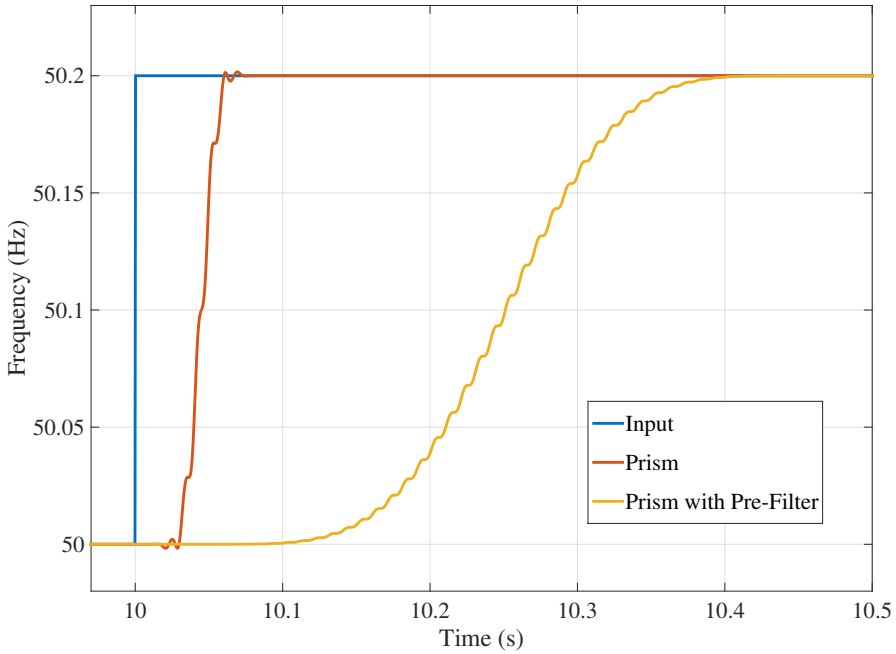


Figure 2.5: The response of Prism algorithm to a step change in input frequency

2.4 CLMS algorithm for result enhancement

2.4.1 Fixed step CLMS enhancement method

From the HTPLL and Prism algorithm described in previous section, an approximate estimate of the voltage frequency in power system can be obtained. However, due to

the transient process of HTPLL and signal distortion, this estimate is not precise, so a complex least mean square (CLMS) section is used to refine the estimated frequency.

In a power system, ignoring the high harmonics and considering the fundamental component, the three-phase voltage at time instant k can be expressed as follows,

$$v_{ak} = v_m \cos(2\pi f k\Delta T + \phi) + \epsilon_{ak}, \tag{2.27a}$$

$$v_{bk} = v_m \cos(2\pi f k\Delta T + \phi - \frac{2\pi}{3}) + \epsilon_{bk}, \tag{2.27b}$$

$$v_{ck} = v_m \cos(2\pi f k\Delta T + \phi + \frac{2\pi}{3}) + \epsilon_{ck}, \tag{2.27c}$$

where v_m is the magnitude of the fundamental component, f is the signal frequency, k is the sampling instant, ΔT is the sampling time, ϕ is the phase of the fundamental component, and ϵ_{ak}, ϵ_{bk}, ϵ_{ck} are the noise in the three-phase voltage data respectively. The complex voltage vector can be derived from the three-phase voltage using $\alpha\beta$-transform with the following format [83, 99]

$$\begin{bmatrix} v_{\alpha k} \\ v_{\beta k} \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} v_{ak} & v_{bk} & v_{ck} \end{bmatrix}. \tag{2.28}$$

The complex expression of the voltage v_k can then be expressed as

$$v_k = v_{\alpha k} + j\, v_{\beta k} = v_n e^{j(2\pi f k\Delta T + \phi)} + \epsilon_k, \tag{2.29}$$

where the magnitude of the complex voltage is v_n, and the error term is ϵ_k. Defining $\hat{v}_k = v_n e^{j(2\pi f k\Delta T + \phi)}$, we then have the expression

$$v_k = \hat{v}_k + \epsilon_k. \tag{2.30}$$

Given the frequency estimation $\tilde{f}_{k-1}$ provided by the HTPLL at time instant $t = k-1$, we then have the following complex voltage model

$$v_k = \tilde{v}_k + \zeta_k = \hat{v}_{k-1}\tilde{w}_{k-1} + \zeta_k, \tag{2.31}$$

where $\tilde{v}_k = \hat{v}_{k-1}\tilde{w}_{k-1}$ is the estimated complex voltage for time instant $t = k$, $\tilde{w}_k = e^{j2\pi\tilde{f}_{k-1}\Delta T}$ is the estimated phase shift of the complex voltage between $t = k-1$ and $t = k$, which is calculated with the approximate estimate made by HTPLLs, and ζ_k is the error between the estimated complex voltage $\hat{v}_k$ and the actual complex

voltage v_k obtained from $\alpha\beta$-transform. In a similar manner to the classical CLMS method, a suitable objective is to minimize the estimation error power [100],

$$E[\zeta_k\bar{\zeta}_k] = E[\zeta_{Rk}^2 + \bar{\zeta}_{Ik}^2] = E[\zeta_{Rk}^2] + E[\bar{\zeta}_{Ik}^2], \tag{2.32}$$

where E designates the expected value and $\bar{\zeta}_k$ is the complex conjugate of ζ_k, ζ_{Rk} is the real component of ζ_k, and ζ_{Ik} is the imaginary component of ζ_k. From (2.31) we have

$$\zeta_k = v_k - \tilde{v}_k = v_k - \hat{v}_{k-1}\tilde{w}_{k-1}. \tag{2.33}$$

So the instantaneous gradient of $\zeta_k\bar{\zeta}_k$ with respect to the real part $w_{R(k-1)}$ of the estimated phase shift vector $\tilde{w}_{k-1}$ is

$$\frac{\partial(\zeta_k\bar{\zeta}_k)}{\partial\tilde{w}_{R(k-1)}} = \zeta_k\frac{\partial\bar{\zeta}_k}{\tilde{w}_{R(k-1)}} + \bar{\zeta}_k\frac{\partial\zeta_k}{\tilde{w}_{R(k-1)}} = \zeta(-\bar{\hat{v}}_{k-1}) + \bar{\zeta}_k(-\hat{v}_{k-1}). \tag{2.34}$$

Similarly, the instantaneous gradient with respect to the imaginary component is

$$\frac{\partial(\zeta_k\bar{\zeta}_k)}{\partial\tilde{w}_{I(k-1)}} = \zeta(j\bar{\hat{v}}_{k-1}) + \bar{\zeta}_k(-j\hat{v}_{k-1}). \tag{2.35}$$

With (2.34) and (2.35), we can apply the steepest descent method to the real and imaginary components of w_{k-1} by adjusting them along their respective negative gradient estimates, which gives

$$w_{Rk} = \tilde{w}_{R(k-1)} - \lambda\frac{\partial(\zeta_k\bar{\zeta}_k)}{\partial\tilde{w}_{R(k-1)}}, \tag{2.36a}$$

$$w_{Ik} = \tilde{w}_{I(k-1)} - \lambda\frac{\partial(\zeta_k\bar{\zeta}_k)}{\partial\tilde{w}_{I(k-1)}}, \tag{2.36b}$$

where $w_k = w_{Rk} + j\,w_{Ik}$ is the refined phase changing vector and λ is the step length. w_k can also be expressed in exponential format with respect to the refined frequency f_{k-1}, which is

$$w_k = e^{j2\pi f_{k-1}\Delta T}. \tag{2.37}$$

and the refined frequency estimate f_{k-1} can be calculated from w_k using (2.37). Combining (2.34), (2.35), and (2.36), we can have

$$w_k = \tilde{w}_{k-1} + \mu\zeta_k\bar{\hat{v}}_{k-1}, \tag{2.38}$$

where $\mu = 2\lambda$ is the design parameter of the algorithm. (2.38) can be used to calculate the enhanced frequency estimate. The calculation needed for the CLMS enhancement algorithm is small, making it suitable for on-line applications.

2.4.2 Adaptive step CLMS enhancement method

The fixed step CLMS enhancement method described in the previous section can improve the initial estimate by correcting it with the detected error term ζ and a design parameter μ. A disadvantage of the fixed step CLMS enhancement method is that the algorithm will always make a correction that is determined by the parameter μ, regardless the amount of the error. When the error term is vary large, the correction determined by μ may not be sufficient and when the error is small, the correction may be too large. Therefore the adaptive step CLMS enhancement method is developed. Based on the variable step size LMS algorithm described in [83], we can make the parameter μ in (2.38) vary with its value determined by the error surface gradient estimate, which is

$$\mu'_{k+1} = \alpha\mu_k + \gamma\bar{\zeta}\zeta, \tag{2.39}$$

with

$$0 < \alpha < 1,\ \gamma > 0, \tag{2.40}$$

and

$$\mu_k = \begin{cases} \mu_{max}, & x > \mu_{max}, \\ \mu_{min}, & x < \mu_{min}, \\ \mu'_{k+1}, & \text{otherwise.} \end{cases} \tag{2.41}$$

where $0 \leq \mu_{min} < \mu_{max}$. The step size is restricted by the upper limit μ_{min} and lower limit μ_{max} and μ_k is always positive. The lower limit is chosen to ensure that the step size will always be larger than 0 and the upper limit is required because an error may be caused by the noise in voltage data, so the correction step should not to be too large. The step size μ_k is controlled by the sized of the prediction error ζ and the parameters α and γ. With (2.39), it can be seen that a large error increases the step size to provide faster tracking, but when the prediction error decreases, the step size decreases as well. In classical LMS algorithms, the step size needs to

be chosen carefully since the algorithm should be bounded. A extremely large μ can lead to the problem that the algorithm does not converge. In our algorithm, however, unlike the choice of step size for classical CLMS algorithm, the CLMS section does not have convergence issue, so the choice of μ_{max} is not limited.

The exponential forgetting factor α must be picked in the range $0 < \alpha < 1$ to provide exponential forgetting performance. A typical value of α that was found to work well in simulation is $\alpha = 0.97$ [83]. The parameter γ is usually very small and should be chosen in conjunction with α and the variance of the data.

For the $\alpha\beta$-transform and the calculation of the estimated complex voltage, the equations used in adaptive CLMS enhancement algorithm are the same as those used in the fixed step method. First of all, the $\alpha\beta$-transform is applied to the three phase voltage signal to obtain a complex voltage vector which contains the phase information

$$\begin{bmatrix} v_{\alpha k} \\ v_{\beta k} \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} v_{ak} & v_{bk} & v_{ck} \end{bmatrix}, \tag{2.42}$$

and

$$v_k = v_{\alpha k} + j\, v_{\beta k} = v_n e^{j(2\pi f k \Delta T + \phi)} + \epsilon_k. \tag{2.43}$$

The error term is calculated from the previous voltage estimate and v_k,

$$\zeta_k = v_k - \tilde{v}_k = v_k - \hat{v}_{k-1}\tilde{w}_{k-1}. \tag{2.44}$$

From the estimate error, the varying step is calculated using

$$\mu_{k+1}^{'} = \alpha\mu_k + \gamma\bar{\zeta}\zeta, \tag{2.45}$$

and

$$\mu_k = \begin{cases} \mu_{max}, & x > \mu_{max}, \\ \mu_{min}, & x < \mu_{min}, \\ \mu_{k+1}^{'}, & \text{otherwise.} \end{cases} \tag{2.46}$$

Finally, with the μ_k obtained from (2.45) and (2.46), the updated frequency estimate term is

$$w_k = \tilde{w}_{k-1} + \mu_k \zeta_k \bar{\hat{v}}_{k-1}, \tag{2.47}$$

and the refined frequency is calculated using

$$w_k = e^{j2\pi f_{k-1}\Delta T}. \tag{2.48}$$

2.5 The combined frequency estimation algorithm

2.5.1 Fixed step CLMS enhancement method

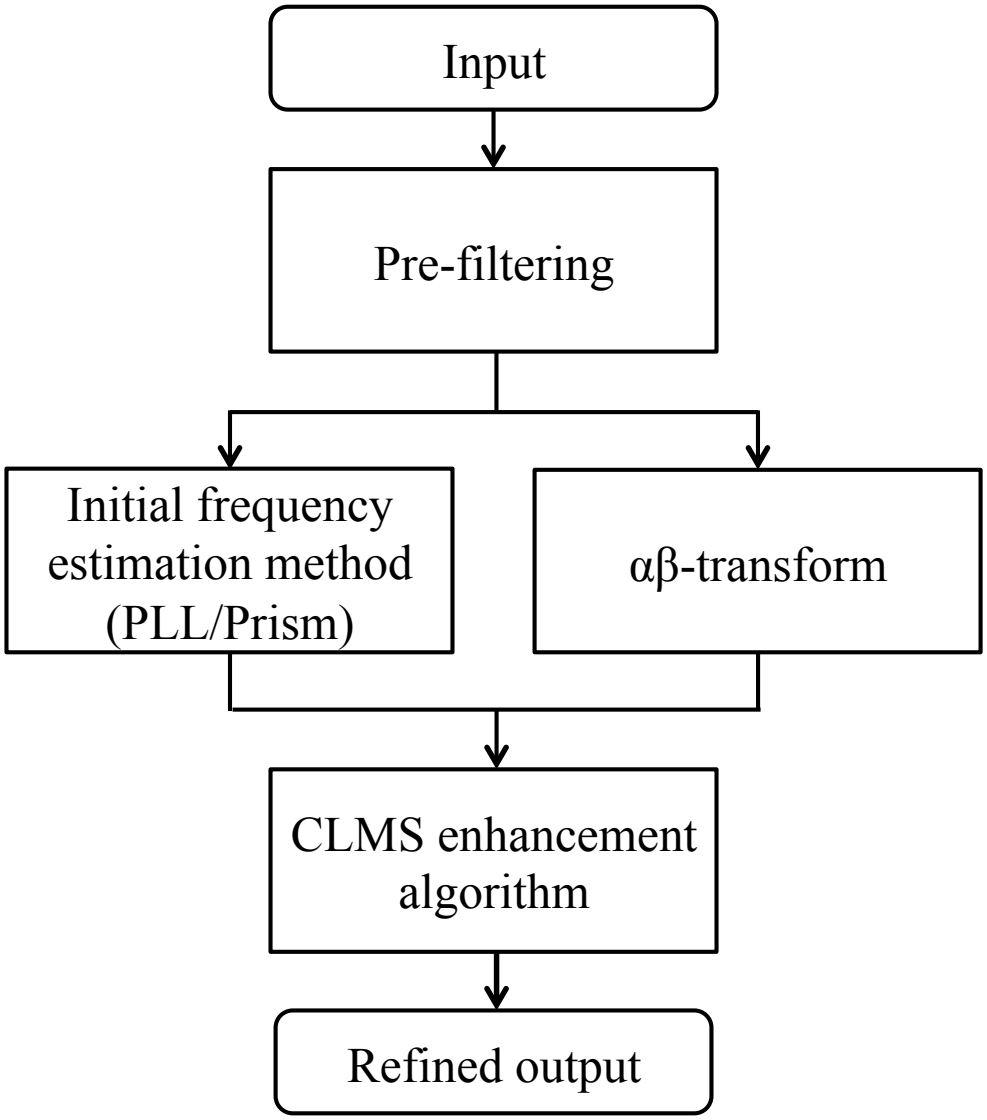


Figure 2.6: Diagram of the fixed step CLMS enhanced frequency estimation algorithm

In the previous sections, the structure of the algorithm and the mathematical details of the algorithm have been explained. In this section, the combined algorithm and way that the different sections of the algorithm coordinate with each other is described.

The flow diagram of the fixed step CLMS enhancement method is shown in Figure 2.6. The input of the combined algorithm is the three phase voltage signal

and the output is the estimated frequency of the input. The three phase voltage input signal can be obtained from the power system by sensors or data logger. Due to the noise and saturation effects, the voltage signals in real power system are distorted sinusoidal waves which contain higher harmonics and noise. These disturbances can result in inaccuracy in the frequency estimation results. Therefore, the data will first be processed by a low-pass or notch pre-filter so that the noise and harmonics can be eliminated.

After being processed by the pre-filter, the comparatively clearer voltage signal will then be sent to the initial frequency estimation section, which can be a well designed PLL or a Prism algorithm. With the filtered input, the initial frequency estimation section can give an estimation of the voltage frequency. However, it worth to be mentioned that this estimation may not be very accurate. If PLL is used as the initial frequency estimation method, because of the transient process, there might be output error when the input is changing; if Prism is taken as the initial frequency estimation method, as it is working with a sliding window, the output is the average result calculated from the all the data contained by the data window. Therefore, in order to obtain the most updated frequency, the result obtained from the initial frequency estimation algorithm should be enhanced. This is why a CLMS enhancement section is needed.

In order to implement the CLMS enhancement algorithm, a complex valued vector containing the information of the input data is needed, which can be obtained by operating an $\alpha\beta$-transform to the three-phase voltage input. As shown in Figure 2.6, this process happens at the same time when the data is processed by the initial frequency estimation method. The modulus of this complex vector is the magnitude of the voltage and the phase of the complex vector is relating to the voltage frequency and time. Phase shift of this complex vector between two consecutive time steps depends on the voltage frequency and sampling time. In practice, the sampling frequency should be a known parameter, which is determined by the sensor or the design of the data logger, so when the phase shift of the vector is obtained, the voltage frequency can be calculated. Unlike the frequency obtained from the

initial frequency estimation method, which may have a delay (transient process of PLL) or represent the average result of a short period (Prism is working with a sliding window), the frequency calculated from the complex vector phase shift is the most updated frequency estimate. During the processing of the algorithm, the input voltage is updated at each time step and processed by the $\alpha\beta$-transform simultaneously, the phase shift of the complex voltage vector between two consecutive time steps is therefore updated at each time interval. So the frequency obtained from the phase shift reflects the most updated frequency. It should be noted that this most updated frequency is calculated simply based on two data points and is very sensitive to disturbance. The noise and higher harmonics can affect the result of this most updated frequency considerably, making it not suitable to be directly used as the frequency result, but it can be used to refined the initial frequency estimate.

When the initial frequency estimate and the complex voltage vector are obtained, these results are input to the CLMS enhancement algorithm. As previously explained, the initial estimate has better tolerance to disturbances but cannot provide the most updated frequency whereas the complex voltage vector indicates the latest frequency but being extremely sensitive to noise. The CLMS algorithm is implemented to refine the initial estimate with the complex voltage vector. It refines the result by minimizing the least mean square value of the frequency estimation error. The refined frequency is therefore not extremely sensitive to noise and can track input frequency change quickly. It is the final output of the combined algorithm. Every time a new data is received, the algorithm will repeat the process described above to generate an estimation of the voltage frequency.

2.5.2 Adaptive step CLMS enhancement method

The flow diagram of the adaptive step CLMS enhancement method is shown in Figure 2.7. The main concept and process of the adaptive step CLMS enhanced frequency estimation algorithm is very similar to that of the fixed step one. The only difference between them is that there is one more section which is used to determine the adaptive step size in the adaptive step algorithm. The adaptive

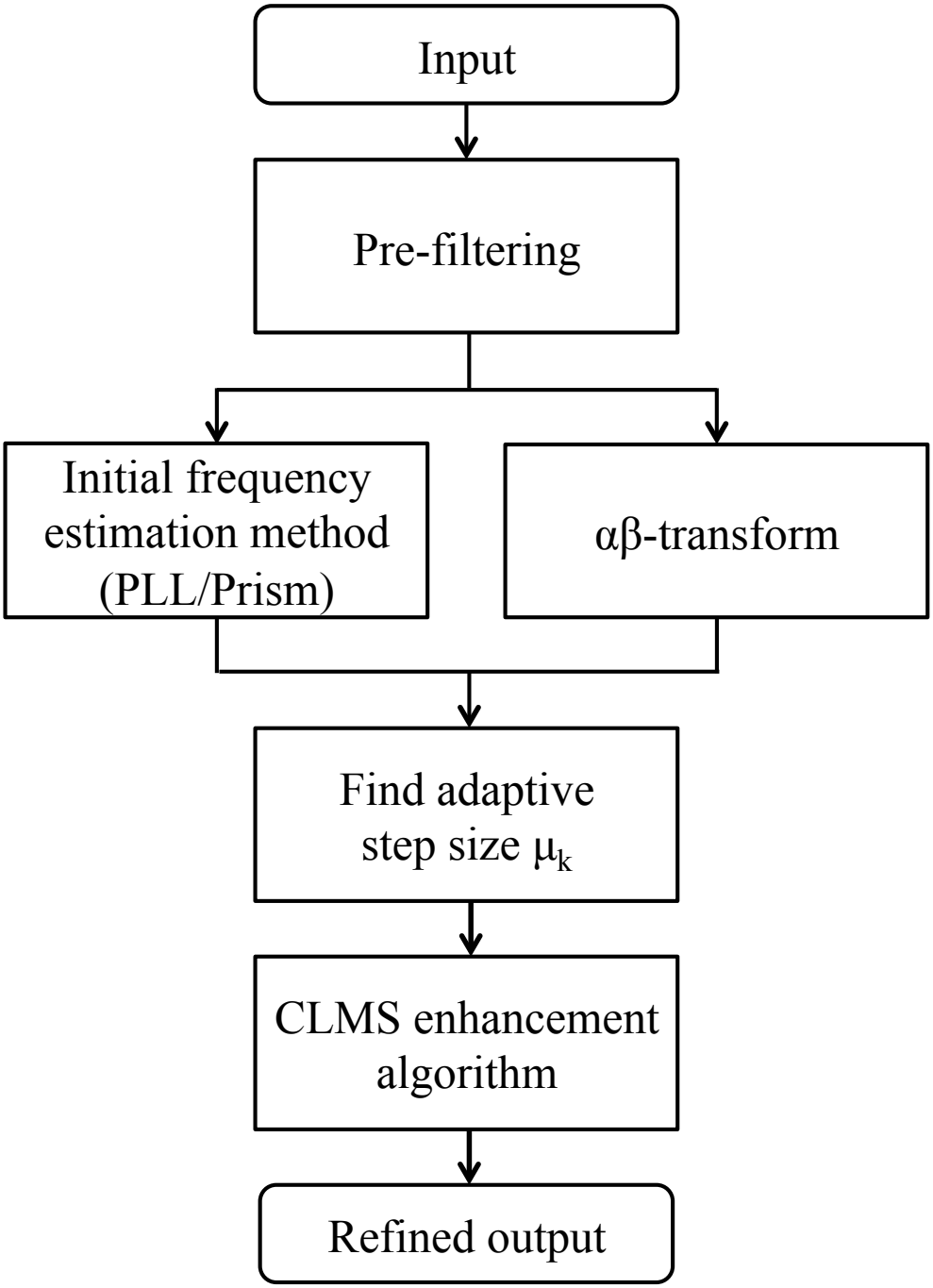


Figure 2.7: Diagram of the adaptive step CLMS enhanced frequency estimation algorithm

step CLMS enhancement method dose not adopt a fixed correction parameter, but instead adjusts its correction parameter according to the estimation error.

After the initial frequency estimate is obtained, combining this result with the complex vector obtained from the $\alpha\beta$-transform, the estimation error can be calculated, so that the adaptive correction parameter μ_k can be updated as well. When the estimation error is comparatively small, it is assumed that the estimation error is more likely to be caused by noise, and the correction parameter tends to be small, so that the refined output is close to the initial output. However, if the estimation error is large, it is more likely that there is variation in the input frequency, so the correction parameter tends to be large and more adjustment will be operated on the initial frequency estimation. In some extreme circumstances, the estimation error can be very large. However in a real power system, the frequency will always change continuously and gradually, such abrupt variation is likely to be caused not by frequency variation but by some unusual disturbances, so an upper limit is set to the correction parameter to avoid overcorrection. After the μ_k is obtained, the data is sent to the CLMS enhancement algorithm and a refined frequency estimation can be obtained. Similarly to the fixed step algorithm, every time a new input data is received, the process in the diagram shown by Figure 2.7 will be repeated, the adaptive step size will be updated and an estimation of the voltage frequency will be generated by the algorithm.

2.6 Simulation and Discussion

In this section, the CLMS enhanced frequency estimating algorithm is tested with simulated data and real data, and the performance of the proposed algorithm is compared with the CLMS approach and the HTPLL [83, 96].

2.6.1 Fixed step CLMS enhanced HTPLL algorithm

Simulated data with step change in frequency

In this case, the performance of the CLMS enhanced HTPLL algorithm is tested using simulated three-phase voltage data. The simulated input data contains a step

change in its input frequency. The sampling frequency of the simulated data is $f_s = 10\text{kHz}$, the magnitude of the input signal is normalised to be $v_{ma} = v_{mb} = v_{mc} = 1$. The frequency of the input signal is $f = 50\text{Hz}$ at the start of the simulation and changes to $f = 50.2\text{Hz}$ at $t = 10\text{s}$. The simulated three-phase data are pure sinusoidal waves without noise. The parameter μ for the CLMS enhanced HTPLL algorithm is set to be $\mu = 0.5$, while the natural frequency of HTPLL is set to be $\omega_n = 4\pi$ for both of the HTPLL in the CLMS enhanced HTPLL algorithm and the HTPLL operating on its own. For the CLMS algorithm operating on its own, the adaptation parameter is set to be $\mu_c = 0.001$

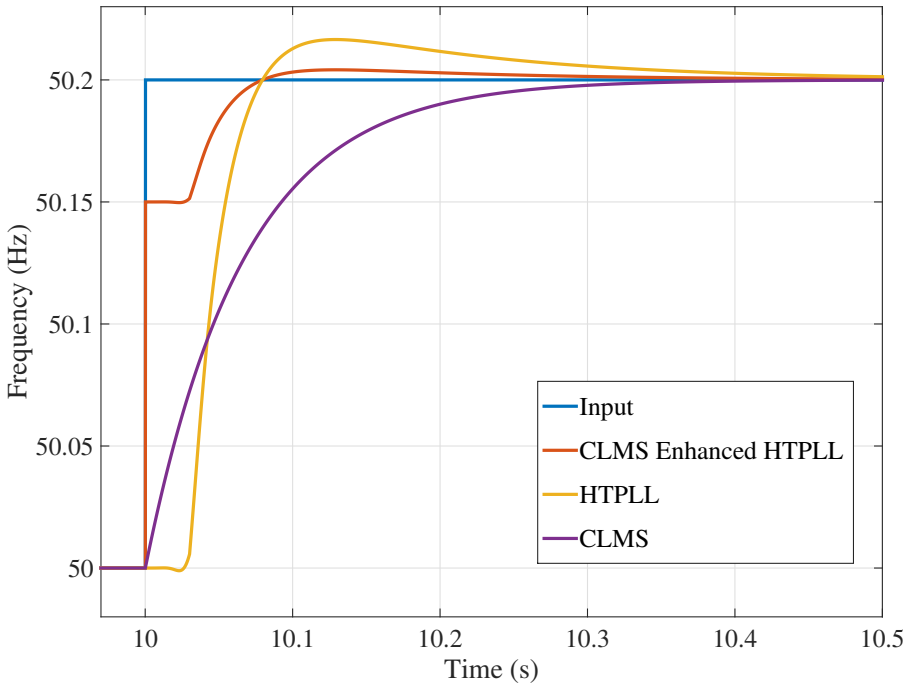


Figure 2.8: Performance of different frequency estimation techniques with a step change in input frequency

The responses of different algorithms to a step change in input frequency are shown in Figure 2.8. In order to compare the performance of different algorithms, the transient time T_t is defined as the time period from the start when change happens till the time when the output stays around the expected output value within 5% of the change amount ($50.2 \pm 0.1\text{Hz}$). After a 0.2Hz step change occurs in

the input frequency at $t = 10$s, the HTPLL reacts to the change with a time delay, which is caused by the Hilbert-transform sliding window, and it worth mentioning that there is an overshoot in the output as well. For the CLMS algorithm, since it does not have sliding window, it reacts to the change quickly and there is no overshoot in the output. However, it takes time for the CLMS algorithm to converge to the correct value and the closer the CLMS output to the input, the slower it moves towards the input frequency. The CLMS enhanced HTPLL algorithm gives the best result among the three algorithms, and it reacts to the change quickly. In the figure, when the input frequency step change occurs at t = 10s, the output of the CLMS enhanced HTPLL immediately reacts to the change and stays at 50.15Hz for a while before converging towards the input. This response is obtained because as shown in Figure 2.1, the input sinusoidal signal is put into both the HTPLL and the $\alpha\beta$-transform section and their outputs are then both input to the CLMS enhancement section. The HTPLL works with a sliding window, which leads to a delay in the output while the $\alpha\beta$-transform section dose not have time delay. So, as the change happens, the HTPLL output stays at the previous value due to the time delay, but the complex vector generated from the input by the $\alpha\beta$-transform already contains the information of the changed frequency. The CLMS enhancement section can detect the difference between the approximate estimate and the actual frequency information contained within the complex voltage vector, and will immediately apply a correction to the approximate estimate with the extent of the refinement being determined by the correction parameter μ. This explains why the CLMS enhancement HTPLL output jumps to around 50.15Hz immediately after the change happens and changes corresponding to the HTPLL output as shown in Figure 2.8. From the figure, the transient time for CLMS enhanced HTPLL is $T_t = 0.06$s, the transient time for HTPLL is 0.25s and that for the CLMS algorithm is $T_t = 0.22$s. The CLMS enhanced HTPLL has the shortest transient time and provides the correct output the fastest. I can also be seen that the output error of the CLMS enhanced HTPLL algorithm is always smaller than that of the HTPLL and CLMS on their own.

Simulated data with sinusoidal frequency change

In this case, the performance of the CLMS enhanced HTPLL algorithm is tested using simulated three-phase voltage data whose input frequency is changing according to a sinusoidal curve. The performance of the proposed algorithm is compared to that of the HTPLL and CLMS approaches.

As the previous test, the sampling frequency of the simulated data is $f_s = 10^4$Hz, the magnitude of the input signal is unified to be $v_{ma} = v_{mb} = v_{mc} = 1$. The frequency of the input signal is changing according to the following rule

$$f = 50 + 0.2\sin(2\pi t). \tag{2.49}$$

The simulated three-phase data are pure sinusoidal wave without noise. The parameters of the algorithms are set to be the same as those mentioned in the previous case. The simulation results are plotted in Figure 2.9.

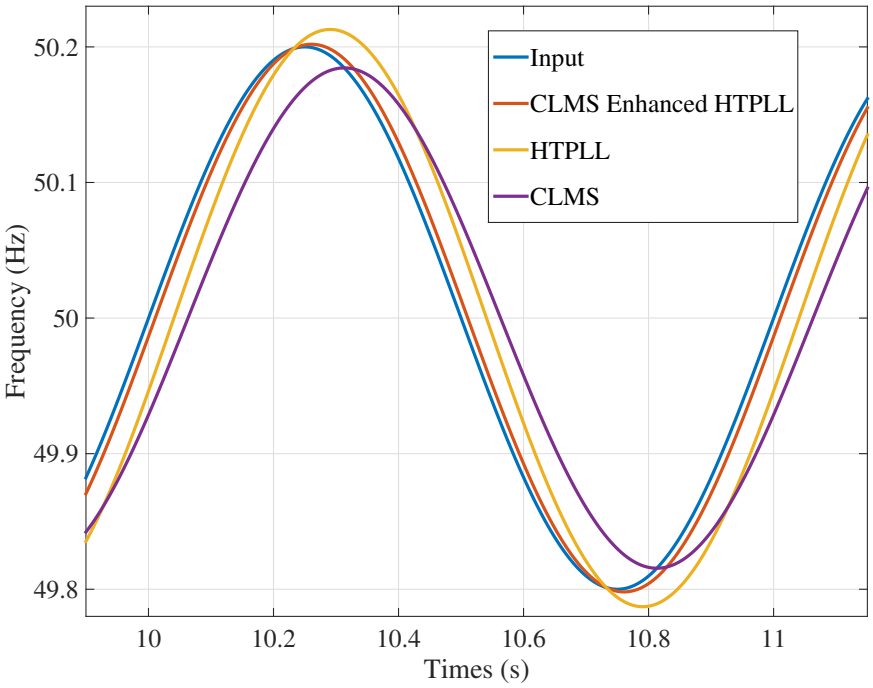


Figure 2.9: Performance of different frequency estimation techniques with changing input frequency

The input frequency variation follows a sinusoidal curve and fluctuating between 49.8Hz to 50.2Hz, which is shown by the blue curve in the figure. The output of

the HTPLL algorithm is changing following the variation of the input, but the fluctuation range of the HTPLL output is larger than that of the input signal as shown by the yellow curve. This is due to the overshoot property of the HTPLL. The CLMS output is also changing according to a sinusoidal pattern, which is shown by the purple curve. As the converging speed of CLMS algorithm is limited by its designed convergence rate μ_c, when the input is constantly changing, its output variation range is smaller than that of the input. The output of the CLMS enhanced HTPLL is the closest to the input signal, which is shown by the red curve. In order to compare the performance of these methods, the corresponding output error is shown in Figure 2.10. In the figure, it can be seen that the error of different algorithms are fluctuating, the frequency estimation error e_f can be defined as the fluctuating magnitude of the error. The error of the CLMS enhanced HTPLL is $e_f = 0.03$Hz, the error of the HTPLL is $e_f = 0.11$Hz and the error of the CLMS algorithm is $e_f = 0.15$Hz. The comparison shows that the estimation error of the CLMS enhanced HTPLL algorithm is smaller than that of the HTPLL or CLMS on their own.

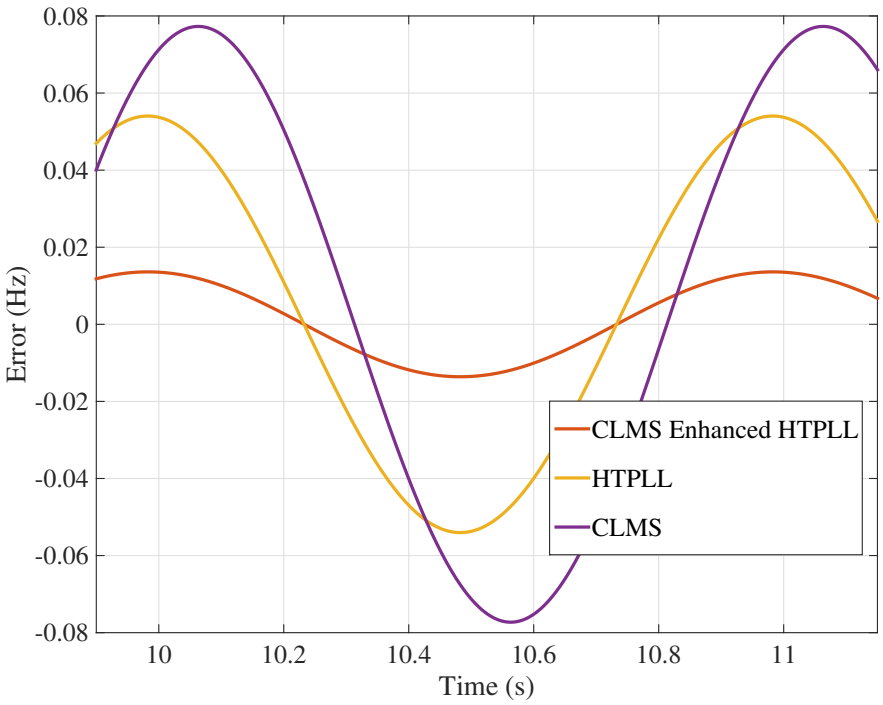


Figure 2.10: The frequency estimation error of different techniques

Real data

In this section, the performance of the algorithms are tested with real three-phase voltage data collected from a power distributed system in Reading.[1] It was obtained by Scottish and Southern Energy (SSE) plc., Reading, and recorded the mains voltage signal of one phase over the period from 12th March 2014 to 14th March 2014 using an energy logger with the sampling frequency of $f_s = 10$kHz. The real periodic voltage is a distorted wave, which contains high harmonics and noise. A section of the input voltage data is shown in Figure 2.11.

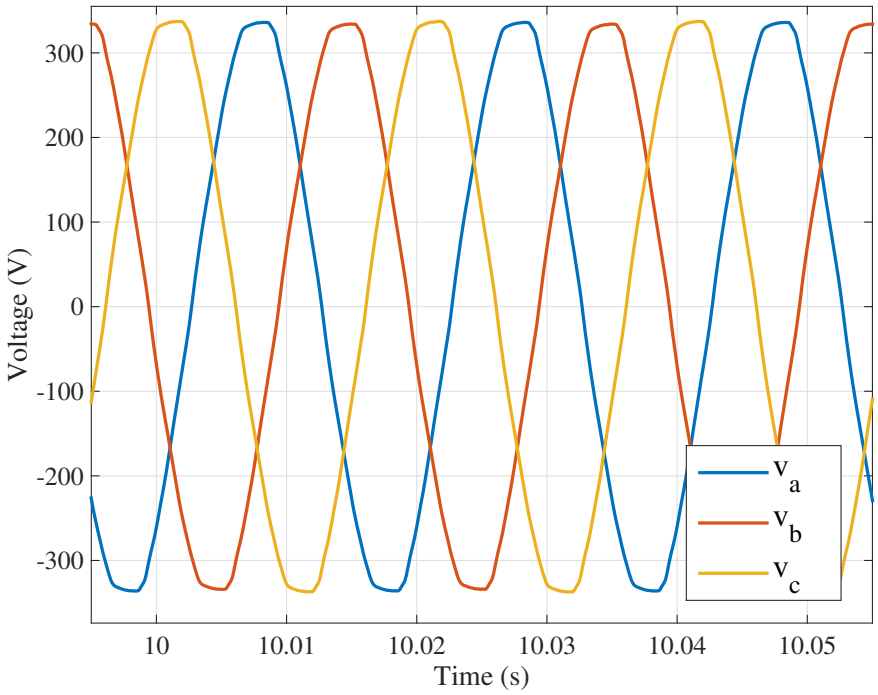


Figure 2.11: Real voltage data

In order to decrease the transient time, the pre-filter of the CLMS enhanced HTPLL algorithm is chosen to be a notch filter, which is implemented using Prism algorithm. The notch filter can eliminate the odd high harmonics. As the frequency value is comparatively stable, the design parameter μ is set to be $\mu = 0.01$ and the natural frequency of HTPLL is $\omega_n = 4\pi$ for both of the HTPLL in the proposed

[1]The real voltage data is provided by Professor Malcolm D. McCulloch and Ren Kang from the Energy and Power Group of the Department of Engineering Science, University of Oxford.

algorithm and the HTPLL operating on its own. For the CLMS algorithm on its own, the adaptation parameter is set to be $\mu_c = 0.001$.

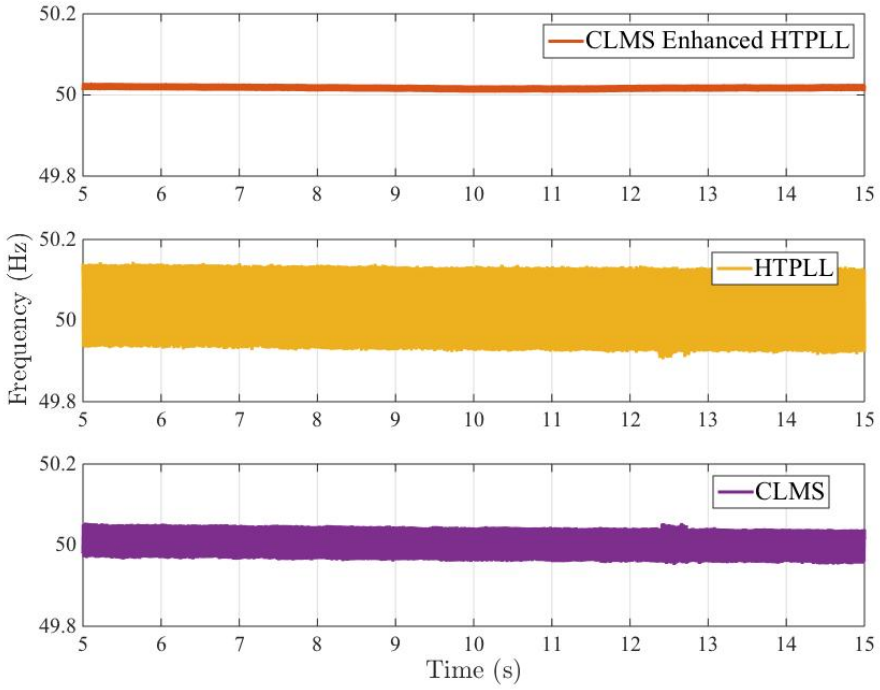


Figure 2.12: Performance of different frequency estimation techniques with real data input

The output of different algorithms are shown in Figure 2.12 and the detail of the output fluctuation is shown by Figure 2.13. With the same real voltage input which contains high harmonics and noise, the output of CLMS enhanced HTPLL output is fluctuating within a range of about $r_f = 0.02$Hz, the HTPLL output is fluctuating within around $r_f = 0.21$Hz while the CLMS output is varying within a range of $r_f = 0.08$Hz. It can be seen that the result of the CLMS enhanced HTPLL algorithm has a lower variation in the frequency estimate.

2.6.2 Fixed step CLMS enhanced Prism algorithm

Simulated data with step change in frequency

In this case, the performance of the CLMS enhanced Prism algorithm is tested using simulated three-phase voltage data, which contains a step change in its input frequency. The set of the simulated input data is the same as that for the previous

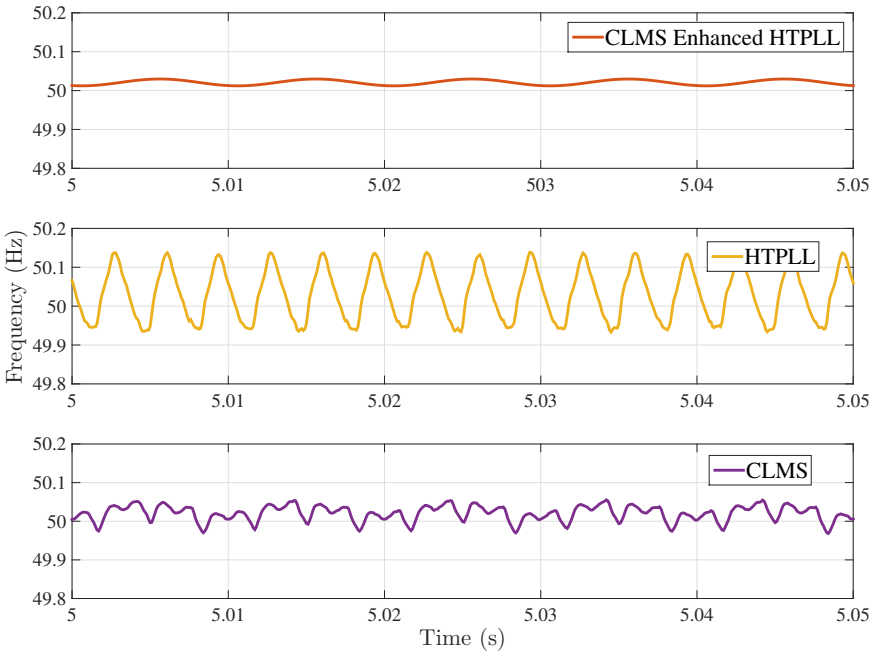


Figure 2.13: Detail of the performance of different frequency estimation techniques with real data input (detail of Figure 2.12)

test. The initial frequency estimation algorithm used here is a Prism recursive signal tracker (RST). In a power system, the voltage frequency is comparatively stable, which is fluctuating around 50Hz, and in order to achieve better performance of Prism, the design frequency of Prism is set to be twice of the input frequency, which is $m = 100$Hz. The simulated three-phase data are pure sinusoidal wave without noise, so any change in the input should occur because of the frequency fluctuation and the correction parameter can be set to be large. The parameter μ for the CLMS enhanced HTPLL algorithm is set to be $\mu = 0.5$. For comparison, the performance of HTPLL, CLMS and Prism on their own are also tested using the same input data. The natural frequency of HTPLL is set to be $\omega_n = 4\pi$. For the CLMS algorithm operating on its own, considering the stability and the convergence rate, its adaptation parameter is set to be $\mu_c = 0.001$.

The reaction of different algorithms to a step change in input frequency is shown in Figure 2.14. In order to further compare the performance of different algorithms, the transient time T_t is defined as the time period from the start when change

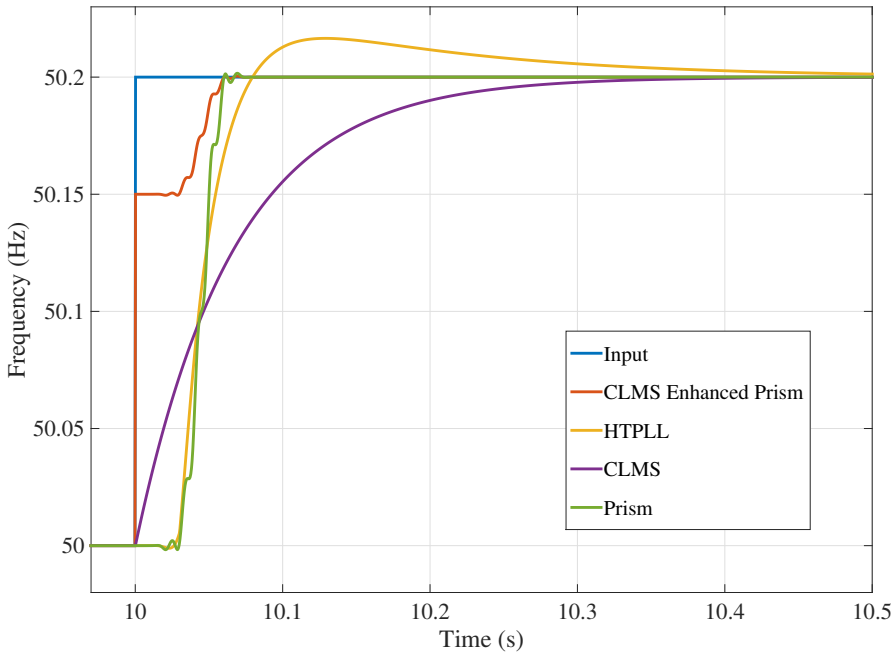


Figure 2.14: Performance of different frequency estimation techniques with a step change in input frequency

happens till the time when the output stays around the expected output value within 5% of the change amount (50.2 ± 0.1Hz). The HTPLL reacts to the input change with a time delay due to the Hilbert-transform sliding window. The transient time T_t for HTPLL to react to the input change of 0.2Hz is $T_t = 0.25$s. The CLMS algorithm shows a comparatively slower converging curve due to the limitation of its convergence rate. The transient process takes CLMS algorithm $T_t = 0.22$s to provide the correct output again. As the Prism algorithm is doing a dual layered integral with a sliding window, it reacts to the input change with a short delay as shown by the green curve. The transient time T_t for Prism is $T_t = 0.06$s. These three methods react to the input frequency change, and give the correct output after a while but the Prism algorithm tracks the input frequency change the fastest.

The CLMS enhanced Prism algorithm output is shown by the red curve in the figure. Comparing to the other three frequency tracking algorithms, it can be seen that it reacts to the input frequency change as soon as the change occurs and it tracks the new input frequency even faster than the Prism. The transient

process of the CLMS enhanced Prism algorithm is about $T_t = 0.05$s, which is the shortest among these methods. In the figure, the combined algorithm reacts to the change quickly since as long as the initial estimate cannot match the input, the CLMS enhancement section will detect the difference and correct the initial estimate according to the error term. Similarly to the response of the CLMS enhanced HTPLL algorithm to a step change, the output of the CLMS enhanced Prism algorithm also stays at around 50.15Hz for a short period. The reason for this response is also similar to that explained in Section 2.6.1, where for the Prism algorithm, it is the Prism sliding window that results in the output delay. In general, the performance of the CLMS enhanced Prism algorithm for a step change in input frequency is better than the other algorithms tested.

Simulated data with sinusoidal frequency change

In this case, the performance of the CLMS enhanced Prism algorithm is tested using simulated three-phase voltage data whose input frequency is changing according to a sinusoidal curve. The performance of the proposed algorithm is compared to that of the HTPLL, CLMS and Prism approaches. The simulated input is the same as the previous test for the fixed step CLMS enhanced HTPLL algorithm. The frequency of the input signal is changing according to the following rule

$$f = 50 + 0.2\sin(2\pi t). \tag{2.50}$$

The simulated three-phase data are pure sinusoidal waves without noise. The parameters of the algorithms are set the be the same as the previous test. The simulation results are plotted in Figure 2.15.

In the figure, all of the outputs vary according to the input frequency fluctuation. The response of the HTPLL and CLMS are the same as those shown in the previous test. The Prism output with changing input is shown by the green curve. It can be seen that unlike the output of HTPLL and CLMS, the Prism algorithm output fluctuation range is almost the same as that of the input, but there is a short delay for the Prism output. In addition, there are some small fluctuation

in the output of Prism algorithm, which is caused by the dual layered integration calculation process of the algorithm.

The fixed step CLMS enhanced Prism algorithm output is shown by the red curve. It can be seen that when the CLMS output is processed by the CLMS enhancement algorithm, the tracking performance is improved and the output delay is shorter.

To compare the performance of these methods, the corresponding output error is plotted in Figure 2.16. The frequency estimation error e_f for the algorithms with sinusoidal input can be defined as the fluctuating magnitude. In this figure, the frequency estimation error of HTPLL is $e_f = 0.11$Hz, the error of CLMS method is $e_f = 0.15$Hz, the error of Prism algorithm is $e_f = 0.12$Hz and the error of the CLMS enhanced Prism algorithm is $e_f = 0.03$Hz. So, it is very clear that among all these algorithms, the output error of CLMS enhanced Prism algorithm is the smallest.

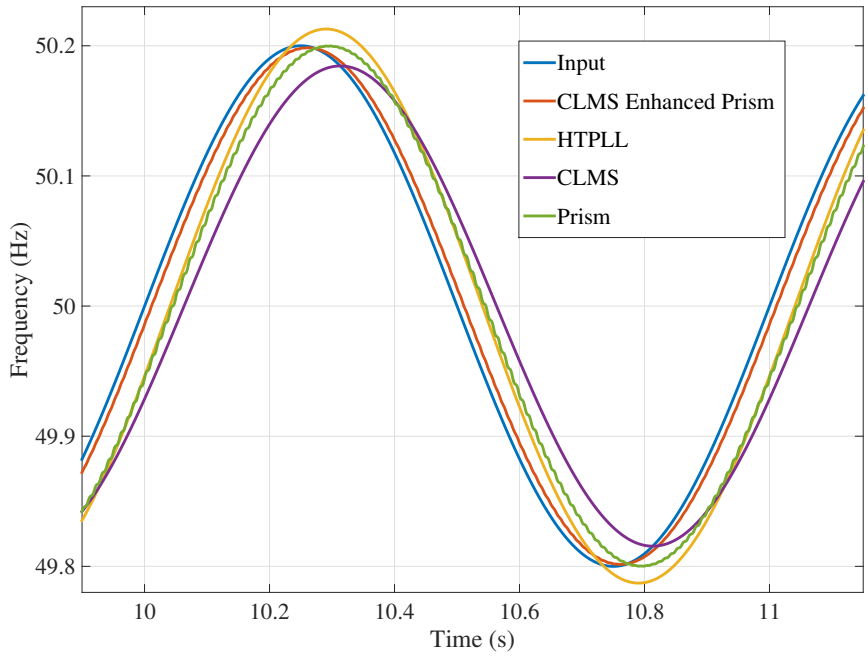


Figure 2.15: Performance of different frequency estimation techniques with changing input frequency

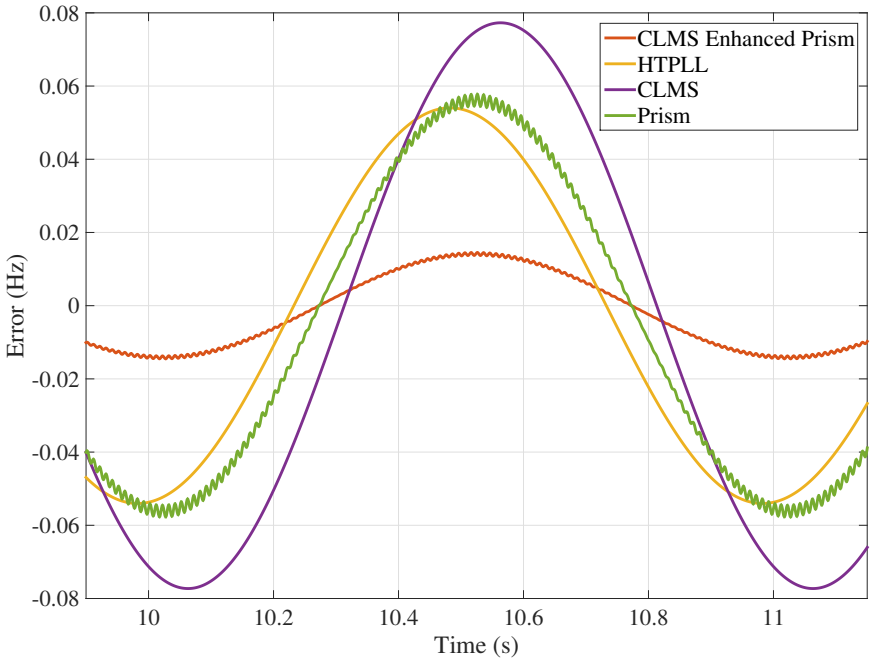


Figure 2.16: The frequency estimation error of different techniques

Real data

In this section, the performance of the CLMS enhanced Prism algorithms are tested with real three-phase voltage data collected from a power distributed system in Reading, and the curve of the input voltage data is shown in Figure 2.11.

In real data, the frequency value is comparatively stable, and the noise in real data is significant. In order to decrease the affect of the noise signal and obtain clearer estimate of frequency, the design parameter μ is set to be $\mu = 0.01$ and the design frequency of Prism is $m = 100$. The parameters for CLMS and HTPLL are set to be the same as the previous real data test. In order to filter the noise data and decrease the transient time, the pre-filter of the CLMS enhanced Prism algorithm is chosen to be a notch filter which is achieved by Prism algorithm. The notch filter can eliminate the odd high harmonics. The output of different algorithms are shown in Figure 2.17 (the detail of Figure 2.17 is very similar to that shown by Figure 2.13). With the same real voltage input which contains high harmonics and noise, the output of CLMS enhanced Prism output is fluctuating within a

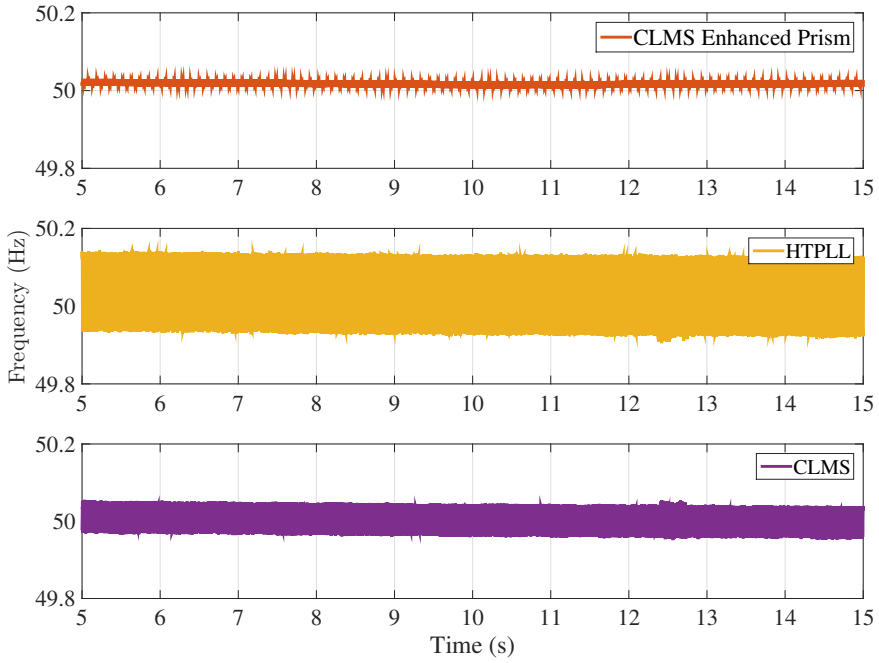


Figure 2.17: Performance of different frequency estimation techniques with real input data (the small spikes in the figure is caused by printing issue)

range of about $r_f = 0.03$Hz, the HTPLL output is fluctuating within around $r_f = 0.21$Hz while the CLMS output is varying within a range of $r_f = 0.08$Hz. It can be seen that the result of the CLMS enhanced HTPLL algorithm has a lower variation in the frequency estimate.

2.6.3 Adaptive step CLMS enhanced HTPLL algorithm

In this section, the adaptive step CLMS enhanced HTPLL algorithm is tested with different simulated and real data.

Simulated data with step change in frequency

In this case, the performance of the adaptive CLMS enhanced HTPLL algorithm is evaluated using simulated three-phase voltage data. The simulated input data is the same as that used for the previous sections. In order to make the algorithm to be more sensitive to the variation of the input data, the forgetting factor α is set to be $\alpha = 0.5$. Considering the scale of the error term and that the input data is a pure sinusoidal signal without noise, the parameter γ is set to be $\gamma = 5 \times 10^9$, the

upper and lower limit of μ_k are set to be $\mu_{max} = 0.5$ and $\mu_{min} = 0.001$ respectively. The natural frequency of HTPLL is set to be $\omega_n = 4\pi$ for both of the HTPLL in the adaptive step CLMS enhanced HTPLL algorithm and the HTPLL operating on its own. For the CLMS algorithm operating on its own, considering the stability of the algorithm and the convergence rate, its adaptation parameter is set to be $\mu_c = 0.001$.

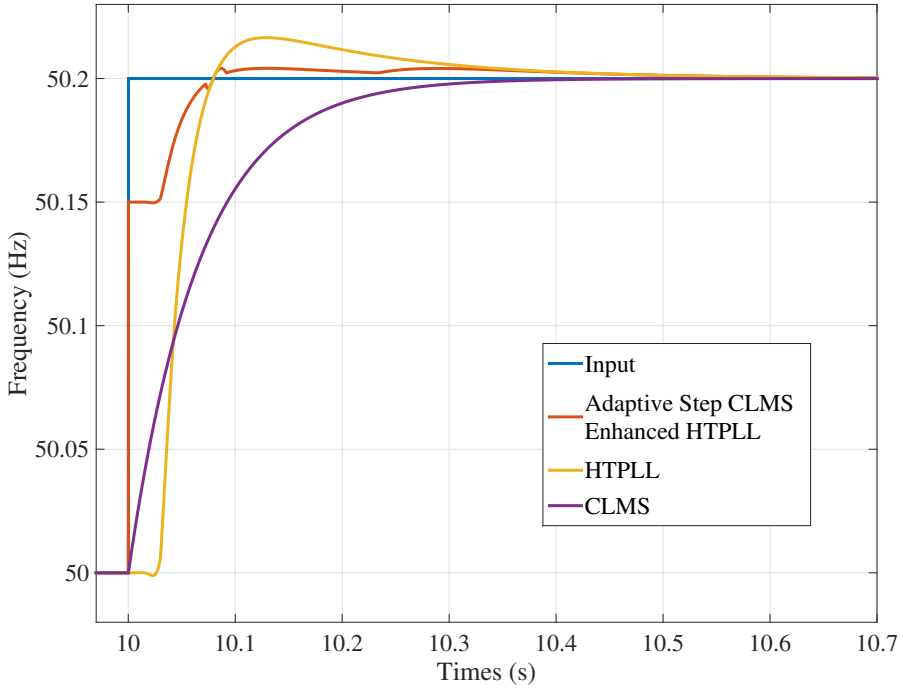


Figure 2.18: Performance of different frequency estimation techniques with step change input

It can be seen in Figure 2.18 that the adaptive step CLMS enhanced HTPLL algorithm can correct the output of the HTPLL output according to the input data. The transient time of HTPLL is $T_t = 0.25$s, the transient time of CLMS algorithm is $T_t = 0.22$s and the transient time of the adaptive step CLMS enhanced HTPLL is just $T_t = 0.06$s. The enhanced algorithm reacts to the input change faster than the other two algorithms. The CLMS enhanced algorithm gives better results compared to that of the HTPLL or CLMS working on their own. Comparing to the result of the fixed step CLMS enhancement method shown in Figure 2.8, it can be seen that the output curve os the adaptive step algorithm is similar to that of the fixed step algorithm, but there are some important differences.

For the fixed step method, the correction parameter is fixed to $\mu = 0.5$, whereas in the adaptive step method, the step size is changing according to the error. The square of the absolute value of the error term ζ is plotted in Figure 2.19 (as ζ is the complex value error of the complex voltage vector, so we plotted the square absolute value here) and the corresponding correcting μ_k is plotted in Figure 2.20. It can be seen that in general, the variation of μ_k follows the change of the error term ζ. The error is quite large the change just happens, which leads to a large correction parameter. The correction parameter μ_k reaches its upper limit $\mu_{max} = 0.5$. Around the time point $t = 10.08$s, due to the overshoot of HTPLL, the initial frequency estimation gets close to the actual input frequency, so the error term decreases and so it is for the μ_k. Then, as the overshoot of HTPLL increases, the error and μ_k becomes larger again and after a while, when the initial frequency estimate becomes closer to the input frequency, the error term and correction parameter μ_k both decrease.

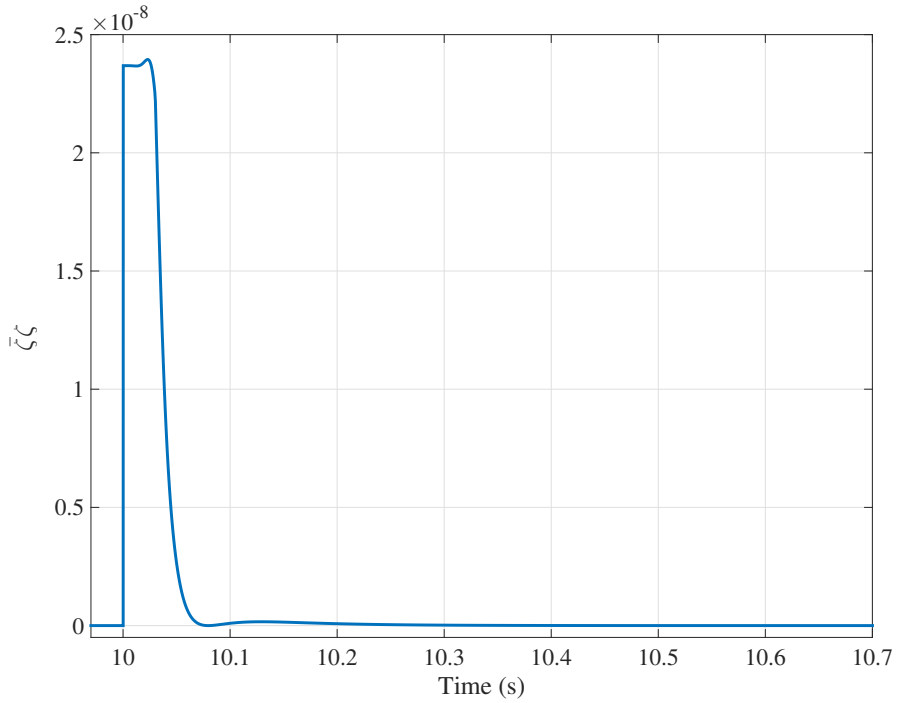


Figure 2.19: Changing of the square of absolute value of the complex error term ζ

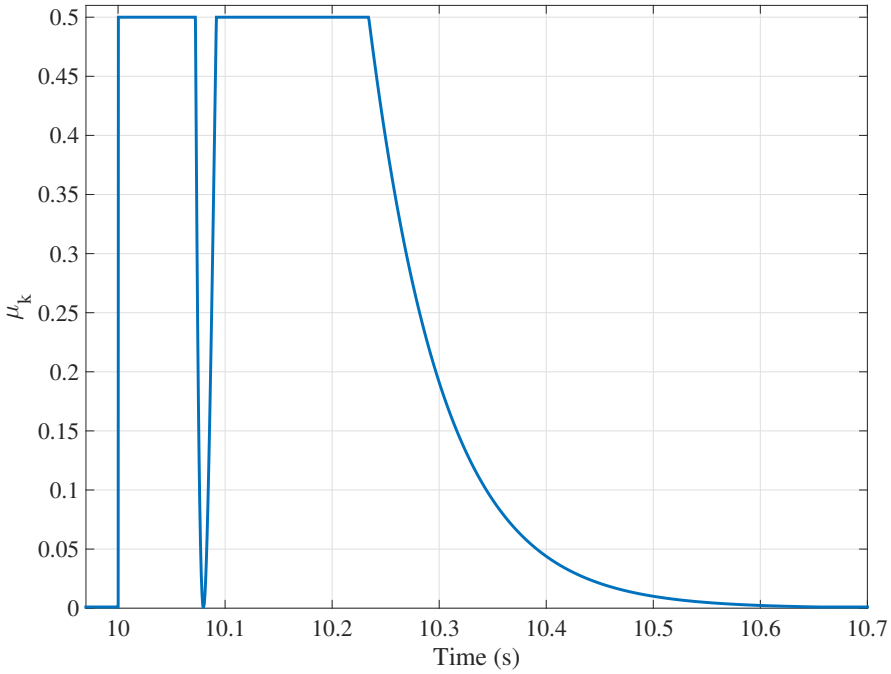


Figure 2.20: Variation of μ_k during the step change

Simulated data with sinusoidal frequency change

In this case, we are going to test the performance of the adaptive step CLMS enhanced HTPLL algorithm using simulated three phase input data with changing frequency.

As before, the simulated input data is still the same. The frequency of the input signal is changing following $f = 50 + 0.2\sin(2\pi t)$. The parameters of the algorithms are set the be the same as the previous case. The simulation result is shown in Figure 2.21

In the figure, the adaptive step CLMS algorithm output is shown by the red curve and it can be seen that the red curve is very close to the input. The estimation error of the HTPLL is $e_f = 0.11$Hz, the CLMS estimation error is $e_f = 0.15$Hz and the error of the adaptive step CLMS enhanced HTPLL is $e_f = 0.3$Hz. The enhanced algorithm provides more accurate result comparing to the other algorithms. The response curve of the adaptive step CLMS enhanced HTPLL output is similar to that of the fixed step algorithm in Figure 2.9. However, it need to be highlighted

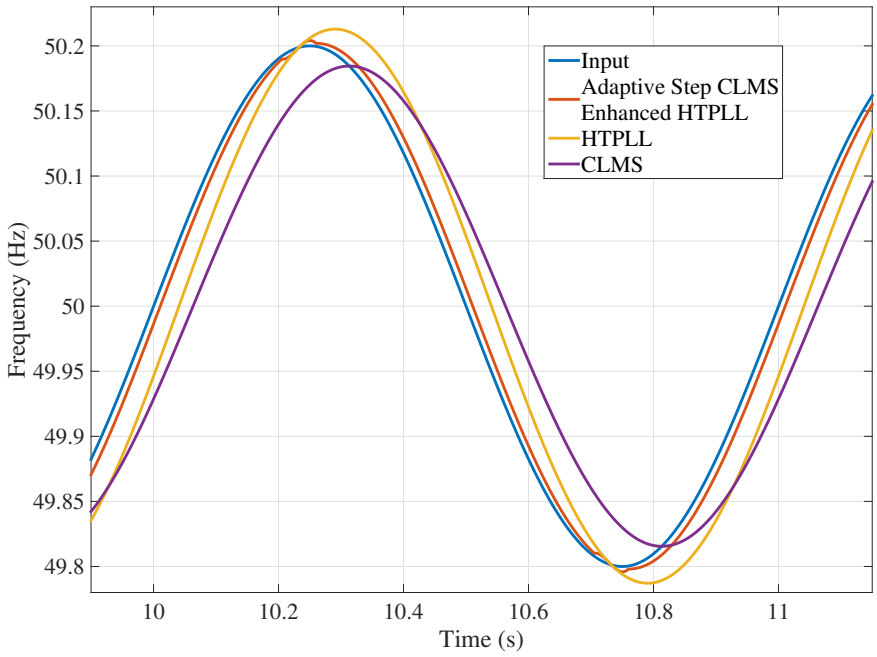


Figure 2.21: Performance of different frequency estimation techniques with changing input frequency

that in the adaptive step algorithm response curve, when the output gets close to the input, the output curve dose not follow a sinusoidal pattern but shows some fluctuation. It is because when the initial frequency estimation is close to the input, the adaptive algorithm's converging parameter μ_k will decrease, so the CLMS enhancement algorithm makes smaller correction to the initial frequency estimate during this period. When the initial estimate error becomes larger, μ_k will increase again. In order to compare the performance of these algorithms better, the corresponding output error is plotted in Figure 2.22. From the figure, it can be seen that the output error of these algorithms are all fluctuating, but the output error for the adaptive step CLMS algorithm is the smallest.

Figure 2.22 shows the variation of the estimated error and the correction parameter μ_k of the adaptive step CLMS enhanced HTPLL algorithm. The figure plots $\bar{\zeta}\zeta$, which is the square absolute value of the complex error term, so its changing frequency is twice the frequency of the output error shown in Figure 2.21. It can also be seen from the figure that when the error term is

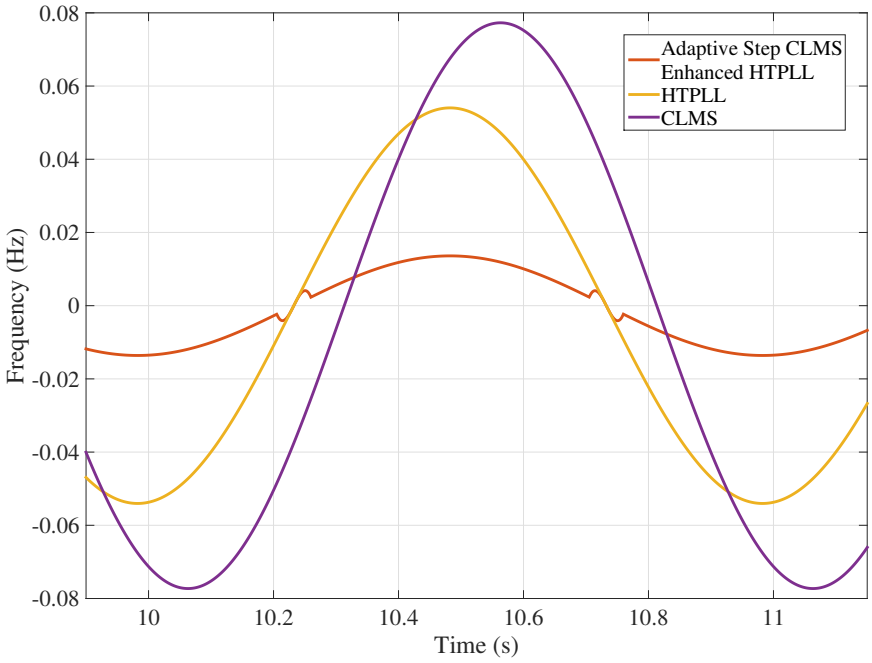


Figure 2.22: The frequency estimation error of different techniques

small, the corresponding μ_k will decrease, making the output of the combined algorithm closer to the output of its initial frequency estimate result obtained from the HTPLL, which explains the output fluctuation of the adaptive step CLME enhanced HTPLL algorithm in Figure 2.21.

Real data

In this section, the performance of the algorithms are tested with real three-phase voltage data collected from a power distributed system in Reading. The curve of the input voltage data is shown in Figure 2.11. In addition, as the real data contains noise signal and high harmonics, the parameter of the adaptive step CLMS enhanced HTPLL is set to be $\alpha = 0.5$, $\gamma = 5 \times 10^4$. The result is shown in Figure 2.24 (the detail of Figure 2.24 is very similar to that shown by Figure **??**).

With the same real voltage input which contains high harmonics and noise, the output of adaptive step CLMS enhanced HTPLL output is fluctuating within a range of about $r_f = 0.04$Hz, the HTPLL output is fluctuating within around $r_f = 0.21$Hz while the CLMS output is varying within a range of $r_f = 0.08$Hz. It

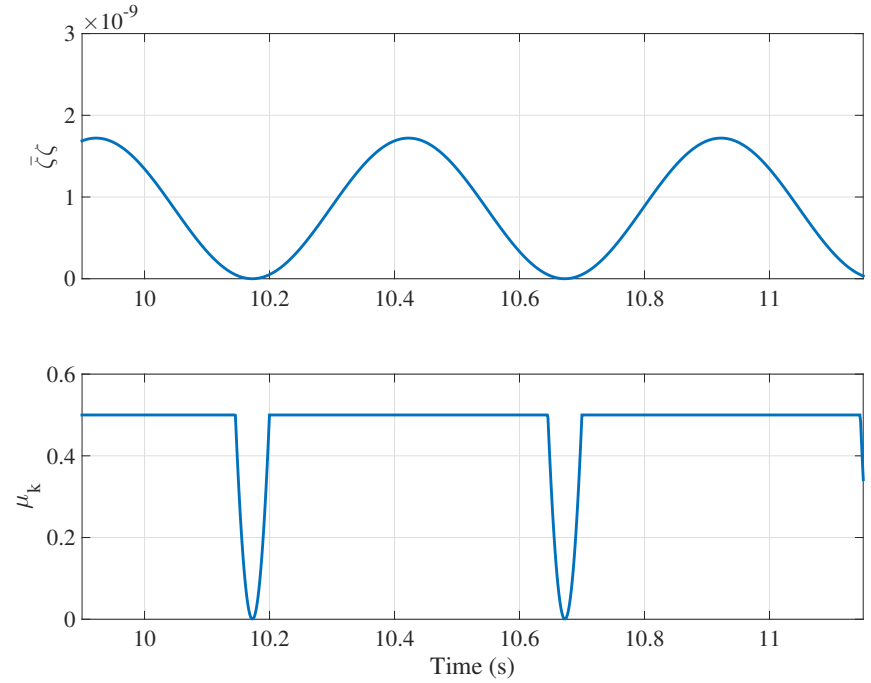


Figure 2.23: The variation of $\bar{\zeta}\zeta$ and μ_k

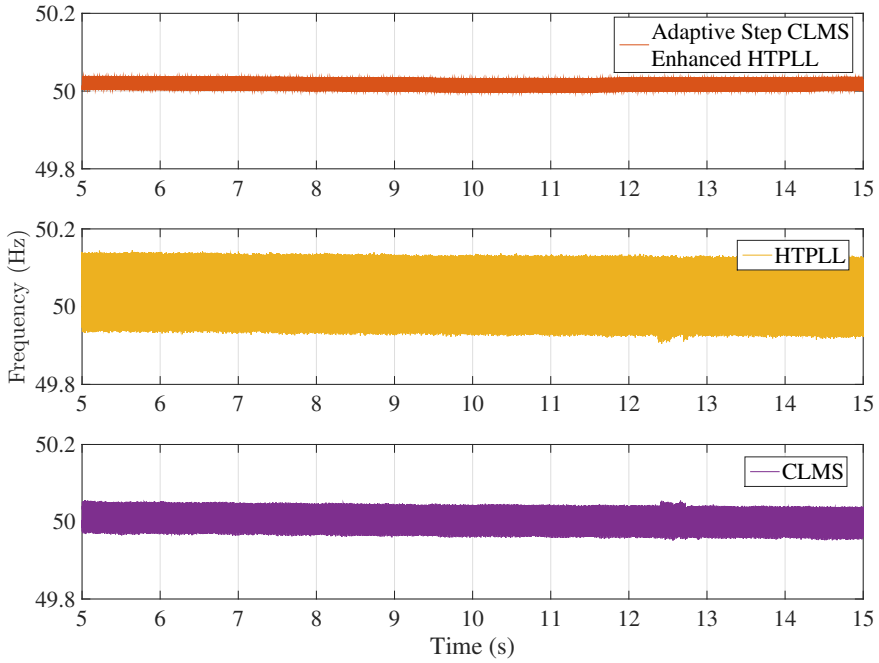


Figure 2.24: Performance of different frequency estimation techniques with real data input

can be seen that the adaptive step CLMS enhanced HTPLL algorithm provide a clearer output comparing to that of the HTPLL or CLMS working alone as it is fluctuating within a smaller range. The output fluctuating range of the adaptive step CLMS enhanced HTPLL is larger than that of the fixed step one. This is because of the difference of the correcting parameter. For real data, the voltage frequency is comparatively stable, the correction parameter is relatively small, so the output fluctuating range will be larger than that of the fixed step algorithm.

2.6.4 Adaptive step CLMS enhanced Prism algorithm

Simulated data with step change in frequency

In this case, the performance of the CLMS enhanced Prism algorithm is tested using simulated three-phase voltage data. The simulated input data contains a step change of 0.2Hz at $t = 10$s in its input frequency.

The algorithm used for tracking the signal is a Prism recursive signal tracker (RST). The design frequency of Prism is set to twice of the input frequency, so $m = 100$Hz. Considering the scale of the error term and the input data is clear sinusoidal signal without noise, the parameter γ is set to be $\gamma = 5 \times 10^9$, the upper and lower limit of μ_k are set to be $\mu_{max} = 0.5$ and $\mu_{min} = 0.001$ respectively. For comparison, the performance of HTPLL, CLMS and Prism on its own are also tested using the same input data. For the HTPLL and CLMS algorithm compared, their parameters are set to be the same as those in the previous tests.

The responses of different algorithms to a step change in input frequency are shown in Figure 2.25. The output of adaptive step CLMS enhanced Prism algorithm is shown by the red curve in the figure. From the figure, the transient time of HTPLL is $T_t = 0.25$s, the transient time of CLMS algorithm is $T_t = 0.22$s, the transient time of Prism working on its own is $T_t = 0.06$s and the transient time of the adaptive step CLMS enhanced Prism is just $T_t = 0.05$s. Comparing to the other frequency tracking methods, it can be seen that it reacts to the input frequency change quickly and tracks the new input frequency even faster than the Prism algorithm. The response of the adaptive step CLMS enhanced Prism algorithm

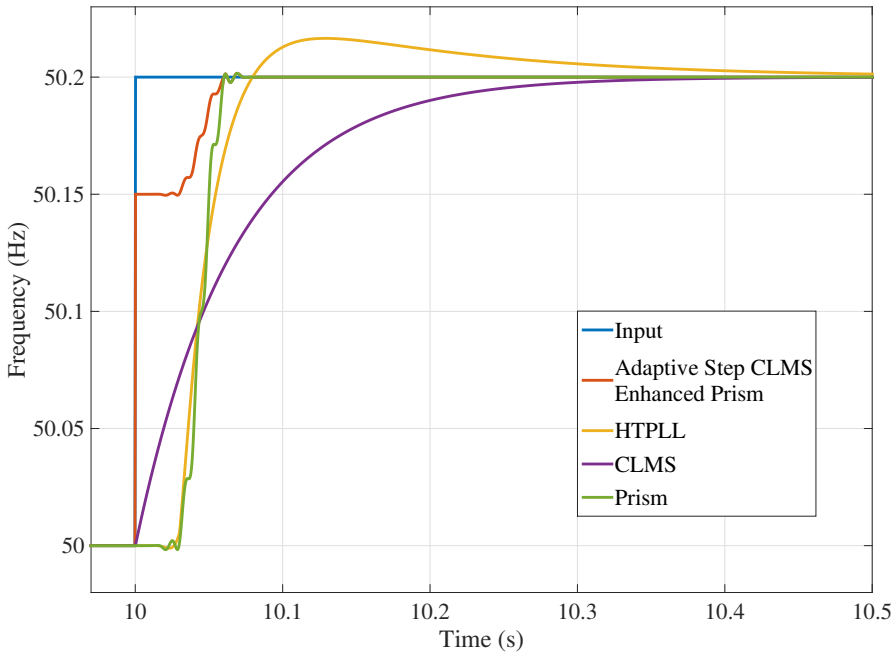


Figure 2.25: Performance of different frequency estimation techniques with step change input

is very similar to that of the fixed step one shown in Figure 2.14. However when the output is close to the input frequency, the converging speed of the adaptive step algorithm will be much slower.

The variation of $\bar{\zeta}\zeta$ and μ_k are plotted in Figure 2.26 and it can be seen that when a frequency change occurs in the input, the error term $\bar{\zeta}\zeta$ increases abruptly, which leads to the increase of the correction parameter μ_k. With a large μ_k, the algorithm is able to converge to the new input frequency quickly. After a while, when the output is close to the new input frequency, $\bar{\zeta}\zeta$ and μ_k both decrease to be very small.

Simulated data with sinusoidal frequency change

The simulated input data and the parameter of the algorithms are the same as the previous tests and the responses are plotted in Figure 2.27.

It can be seen that all the outputs are changing following the variation of the input. The response of the adaptive step CLMS enhanced Prism algorithm, which is shown by the red line, is very close to the input. Due to the property of the adaptive

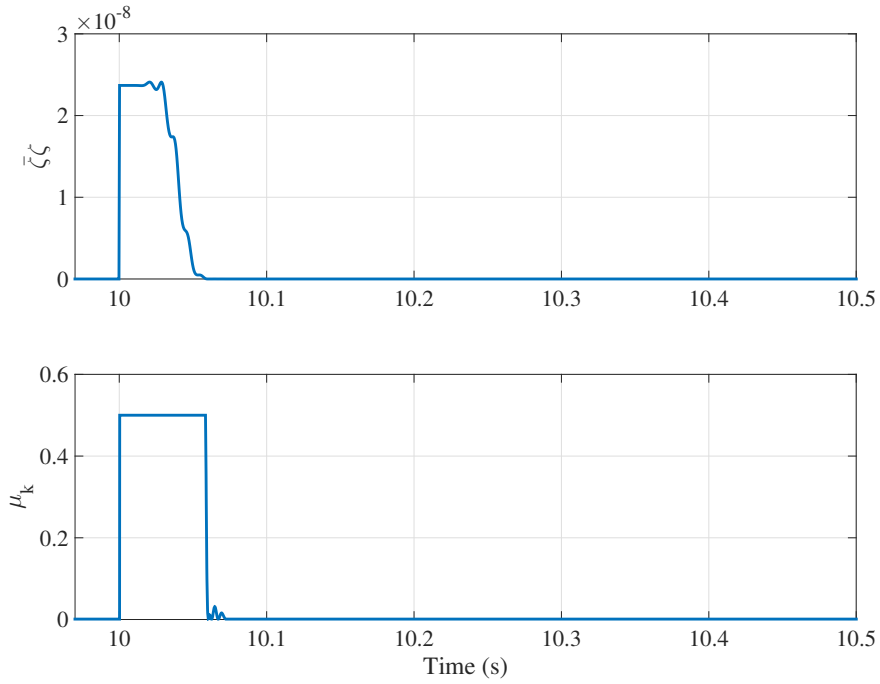


Figure 2.26: The variation of $\bar{\zeta}\zeta$ and μ_k

step algorithm, when the HTPLL output gets close to the input, the adaptive converging parameter μ_k will become smaller, so there is a fluctuation in the adaptive step algorithm output at around $t = 10.2$s. The corresponding output error is plotted in Figure 2.22. From the figure, it can be seen that the output error of the adaptive step CLMS enhanced Prism algorithm is $e_f = 0.03$Hz, the error of Prism is $e_f 0.12$Hz, the output error of HTPLL is $e_f = 0.11$Hz and the error of CLMS is $e_f = 0.15$Hz. The output error of the adaptive step enhanced algorithm is the smallest.

The corresponding $\bar{\zeta}\zeta$ and μ_k are plotted in Figure 2.29, from the figure, the error term changes periodically, leading to the periodic change of the adaptive step size μ_k.

Real data

In this section, the performance of the algorithms are tested with real three-phase voltage data. Considering the noise and high harmonics in the real data, the parameter of the adaptive step CLMS enhanced Prism algorithm is set to be $\alpha = 0.5$, $\gamma = 5 \times 10^4$. A section of the input data is shown in Figure 2.11. The

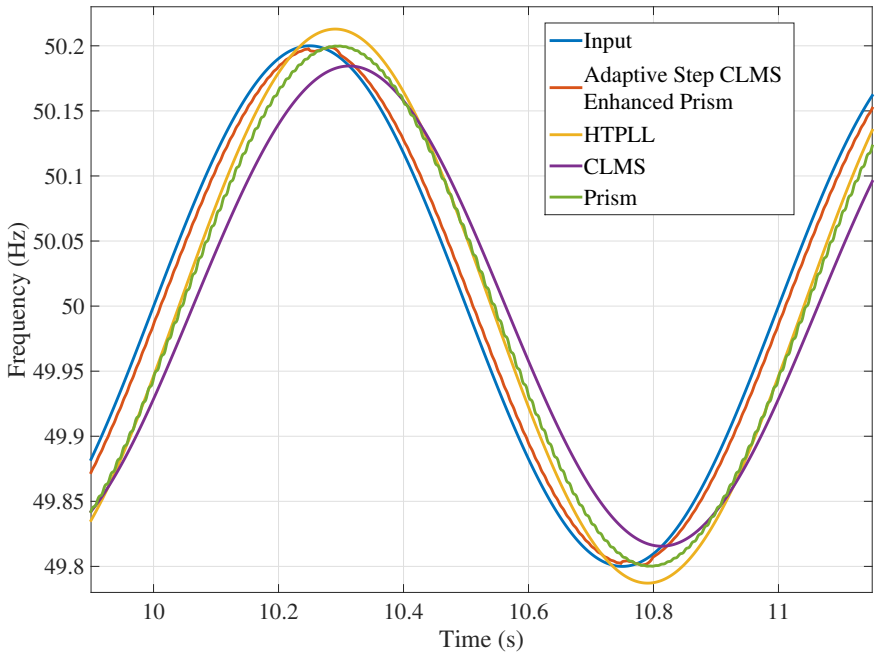


Figure 2.27: Performance of different frequency estimation techniques with changing input frequency

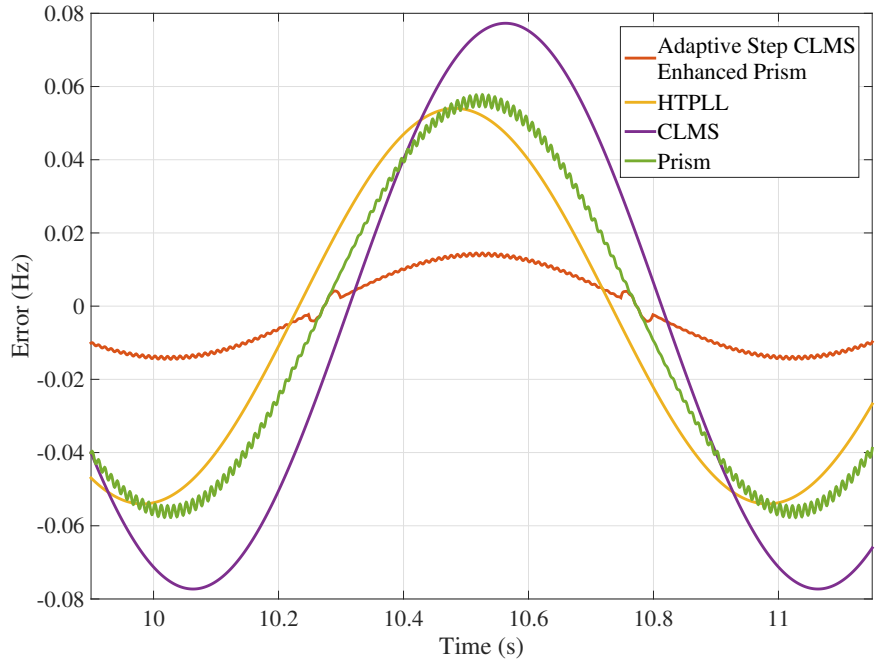


Figure 2.28: The frequency estimation error of different techniques

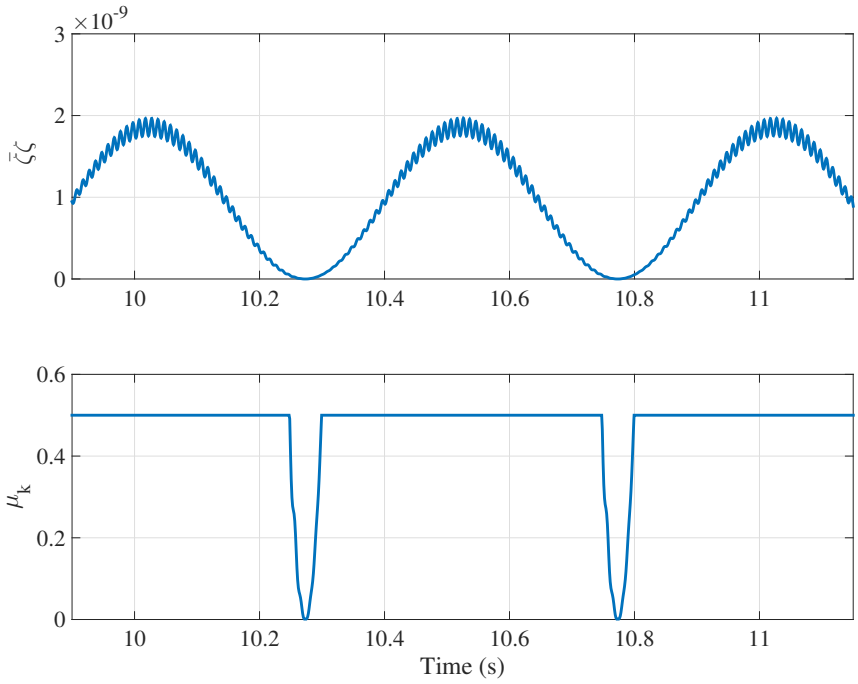


Figure 2.29: The variation of $\bar{\zeta}\zeta$ and μ_k

response of the different algorithms are shown in Figure 2.30, and the detail of Figure 2.30 is shown in Figure 2.31.

With the same real voltage input which contains high harmonics and noise, the output of adaptive step CLMS enhanced Prism output is fluctuating within a range of about $r_f = 0.04$Hz, the HTPLL output is fluctuating within around $r_f = 0.21$Hz while the CLMS output is varying within a range of $r_f = 0.08$Hz. It can be seen that the result obtained from the adaptive step CLMS enhanced Prism algorithm fluctuates in a smaller range comparing to that obtained from the HTPLL or CLMS algorithms. However, the output error of adaptive step CLMS enhanced Prism algorithm is larger than that of the fixed step one, which is similar to the condition of the enhanced HTPLL algorithm. The reason is similar as well. When the adaptive step algorithm is adopted, the correcting parameter tends to be relatively small when there is no abrupt change in the input, so the fluctuating range of the algorithm's output will be slightly larger than that of the fixed step algorithm.

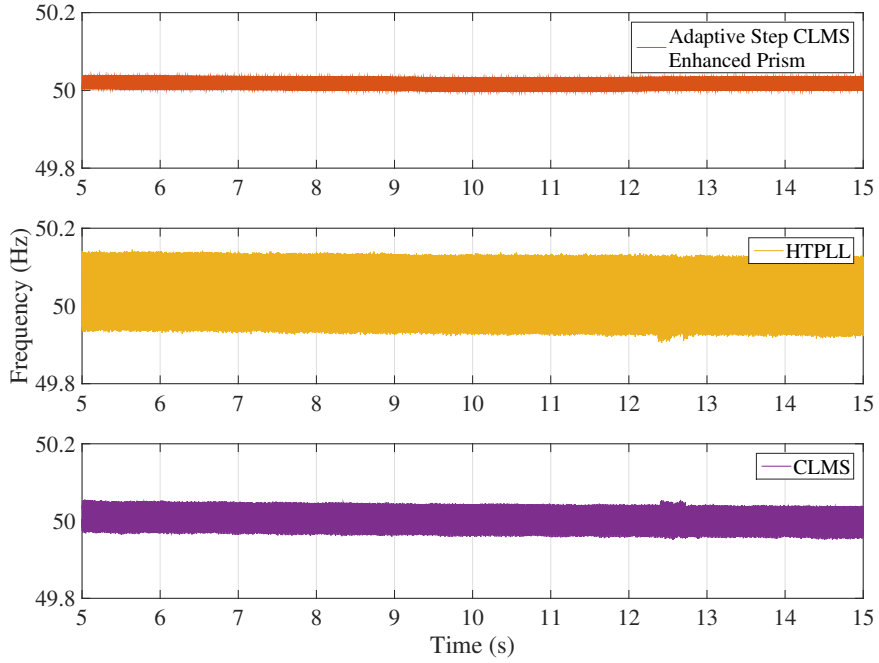


Figure 2.30: Performance of different frequency estimation techniques with real data input

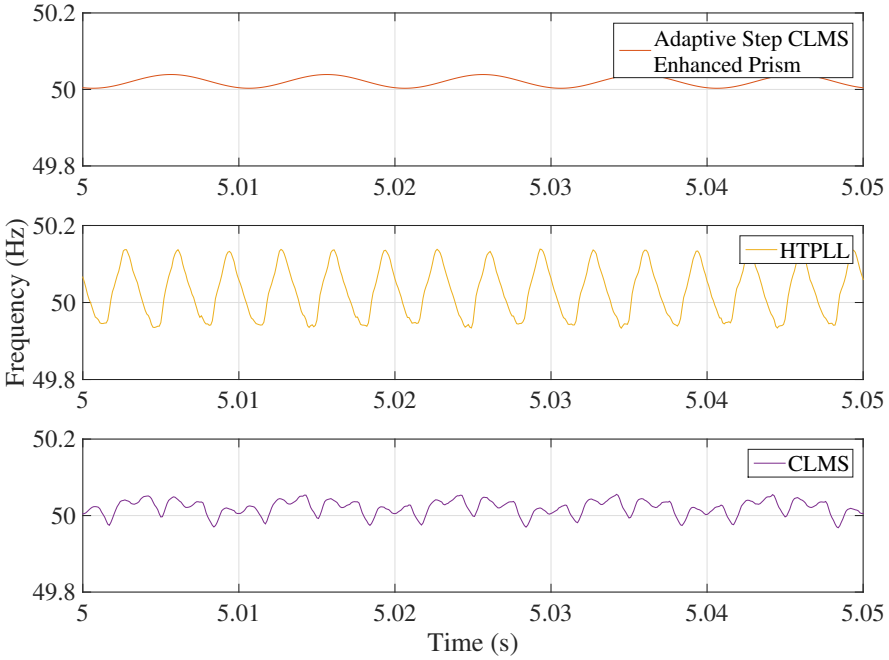


Figure 2.31: Detail of the performance of different frequency estimation techniques with real data input shown in Figure 2.30

2.7 Conclusion

In this section, a new approach for power system frequency estimation is presented. The algorithm takes the three-phase voltage as the input and Prism pre-filter is used to reduce the high harmonics and noise. Initial frequency estimating techniques are used to obtain an approximate estimate of the frequency, then the initial estimate is refined by a CLMS section. The complex voltage vector that being input to the CLMS section is obtained from an $\alpha\beta$-transform.

The Prism pre-filter decreases the noise in the input and the HTPLL or Prism algorithm provides the algorithm with good initial frequency estiamte. The CLMS section then refines the approximate estimate and improves the accuracy. According to the simulation results, when the input frequency undergoes a step change or is changing constantly, the performance of the proposed algorithm is better than that of the HTPLL or CLMS working on their own. Real data collected from the power system is also used for testing the proposed algorithm and it shows that the proposed CLMS enhanced method can provide clear frequency estimation result for real three-phase voltage data.

The HTPLL and Prism algorithms can both be used as the initial frequency estimation technique for our algorithm, and the results are presented in the simulation result section. The HTPLL is a classical method, which has been well developed and been proved that it can provide good dynamic performance. Besides, the phase detector and loop filter in PLL can filter the noise and high harmonic signal in the input efficiently, but the HTPLL's reaction speed is limited by its Hilbert-transform sliding window. Comparing to the disadvantages of HTPLL, the Prism algorithm is working with a shorter sliding window and can track the input frequency effectively, but it is sensitive to noise in the input. So when implementing Prism as the initial frequency estimation technique, it is imperative that a pre-filter should be added. In practice, the dynamic performance of the initial frequency estimating methods can be improved by adjusting their design parameters. For the HTPLL, by improving the system nature frequency ω_n, the transient time can be decreased and therefore it can react to input frequency change more quickly.

For Prism method, as its dynamic performance is relating to the length of its sliding window, we can pick a larger Prism frequency m. This will lead to a smaller sliding window and the dynamic performance can be improved. But it needs to be mentioned that changing the Prism frequency may change the error reduction factor of the algorithm, which may affect the algorithm convergence rate.

In this algorithm, the choice of the parameters of the CLMS enhancement section is also important. For the fixed step CLMS enhancement method, a large step parameter μ can make the system more sensitive to input changes, but will make it more sensitive to noise signal as well. Therefore, when implementing this method to real data, the variation of the input needs to be taken into consideration. From the test with the real data used in this thesis, $\mu = 0.01$ would be a good choice. However, when the input is a clearer signal with less noise, a larger μ can be used and this can improve the dynamic performance of the algorithm. For the adaptive CLMS enhancement method, the adaptive parameters are also needed to be pick carefully. For the real data used in our research, $\alpha = 0.5$ and $\beta = 5 \times 10^4$, can be a good choice. As a pre-filter is used in the algorithm, so when using the adaptive CLMS enhancement algorithm to detect other real input data, even though the input data may include more noise, the parameters proposed should also work.

Numerical examples provided in the result and simulation sections show that the proposed enhanced algorithm provide better performance than the traditional methods. Compared to the traditional methods, the enhanced frequency estimation method can track the input frequency variation with better accuracy and can react to input changes faster.

3

Automatic detection of tap changes

Contents

3.1 Introduction **71**
3.2 Algorithm structure **73**
3.3 Frequency tracking technique **75**
3.3.1 Classical PLL 75
3.3.2 Hilbert-transform PLL 80
3.4 Recursive LMS algorithm **83**
3.4.1 Classical recursive least square algorithm 84
3.4.2 QR decomposition windowed recursive least square algorithm 90
3.5 Hypothesis test **94**
3.5.1 Generalized likelihood ratio test 95
3.5.2 Two sample t-test 99
3.6 Simulation and discussion **102**
3.6.1 PLL result with real input 102
3.6.2 QRD-WRLS algorithm result with real input 104
3.6.3 Hypothesis test result 109
3.6.4 GLR test result 110
3.6.5 Two sample t-test result 114
3.7 Conclusion **117**

3.1 Introduction

Demand side management (DSM) is a key concept for the successful implementation of smart grids, as it modifies consumers' demand for energy via a range of methods,

including financial incentives [101], to improve the efficiency of the power network. For example, during peak hours, users can be encouraged to turn off appliances, such as washing machine or air conditioning units, and to turn them on again during off-peak period. Users can also use energy storage units to store energy during off-peak hours and discharge them during peak hours.

In a power system, the supply and demand are constantly changing, so a mismatch always exists. The stress in a power system is defined by how close the demand is to the capacity of the operating limit [102] and such a stress exists at both the national level and local level. At the national level, the imbalance between the generator and load will lead to the variation in the frequency of the sinusoidal voltage signal, whereas at local level, the stress will result in voltage variation in power distribution networks [103, 104]. In this section, we will concentrate on determining the local network stress from the variation of voltage at the demand side.

Currently, much research has been done on identifying the local network stress, however most stress identification methods rely heavily on communication techniques such as ZigBee, WIFI or 3G/LTE [2, 105]. In these implementations, information from users is sent to a data concentrator and processed, so that the network stress information will be transmitted from the control centre to users. Implementing such a huge amount of communication requires a well-developed information network, which is both costly and vulnerable.

In a power distribution system, in order to balance the voltage loss along the cable and keep the voltage at the user side stable, on-load tap changer (OLTC) transformers are used. An OLTC transformer reacts to different load conditions by moving a tap to a different winding contact, adjusting its transformer ratio and changing its output voltage. This is referred to as a tap change and tap changes occur according to the variation of local network stress. Since the effect of tap changes can be detected by measuring the change in magnitude of the 230V voltage signal, which is accessible to the users and this can be exploited to allow local users to detect changes in the stress of the network. The benefit of

identifying network stress by detecting tap changes is that communication with the data concentrator is not required.

3.2 Algorithm structure

Tap changes are implemented by on load tap changer (OLTC) transformers which are located at the power transmission network. The voltage variation caused by tap changes can be detected at the demand side as a variation in the magnitude of the voltage signal. However, even when there is no tap change, due to the variation in the load current and other disturbances, there will still be voltage magnitude variations. Therefore, in order to accurately separate tap change from the noise and other disturbances, we need a hypothesis test in the algorithm to distinguish tap changes from other kinds of voltage fluctuations. The magnitude of the voltage of the sinusoidal signal is estimated using a least mean square algorithm. Additionally, in order to obtain a precise magnitude of the voltage, the least mean square algorithm needs to know the frequency of the voltage signal, so a frequency tracking component is also required by the algorithm.

From the discussion above, the approach used here is to combine three main components in the automatic tap change detection algorithm which are: Phase-locked Loop (PLL) for frequency estimation, Windowed Recursive Least Mean Square (WRLMS) algorithm for magnitude calculation and Hypothesis Test for change detection. The input signal of the algorithm is the voltage signal at the load side and the algorithm should return the result when a tap change occurs. The structure of the algorithm shown in Fig. 3.1.

The input of the tap change detection algorithm is taken to be the sinusoidal voltage signal obtained form the load side of the power distribution system. In practice the voltage signal contains higher harmonics and the effect of these harmonics on the detection algorithm will be discussed below. In the UK, the standard of the (root mean square) voltage at the user side is 230V±10% and the

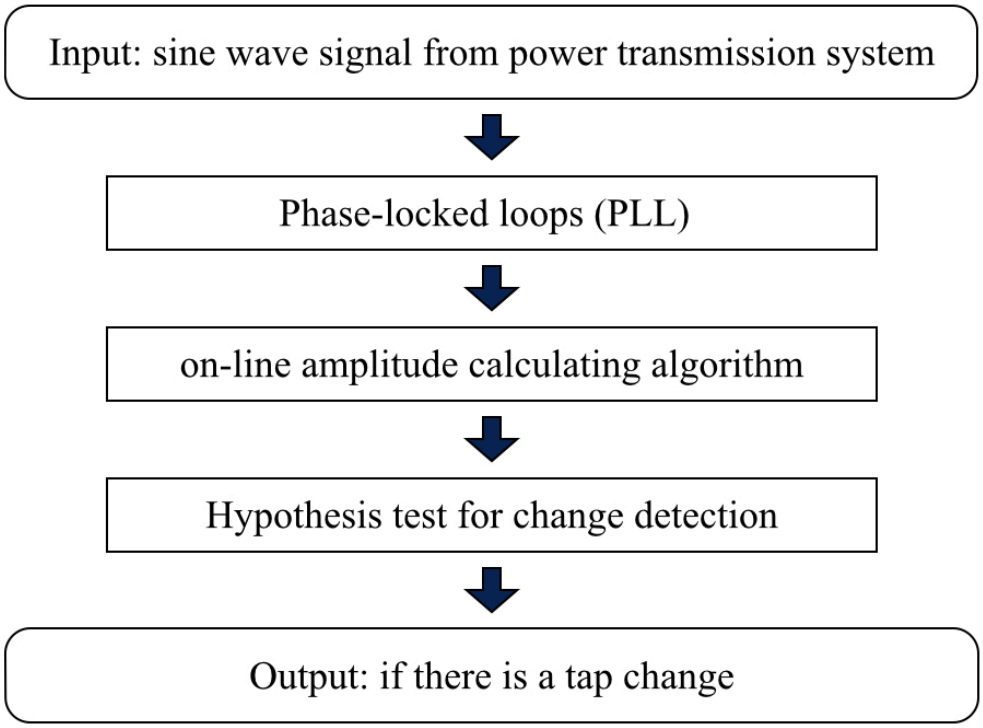


Figure 3.1: Algorithm structure of the tap change detection diagram

permitted range of the frequency is 50Hz±1%. At any given time t, voltage of the fundamental component $y_1(t)$ can be expressed as follows

$$y_1(t) = A_1 \cos\phi_1(t), \tag{3.1}$$

where A_1 is the magnitude and $\phi_1(t)$ is the phase. The initial phase at time $t = 0$ is $\varphi_1(0) = \theta_0$. If we know the voltage frequency $f_1(t)$ at any given time t, then equation (3.1) can be written as

$$y_1(t) = a_1 \cos(\varphi_1(t) + \theta_0) + b_1 \sin(\varphi_1(t) + \theta_0), \tag{3.2}$$

where $\varphi_1(t) = 2\pi \int_0^t f_1(s)ds$ and $A_1 = \sqrt{a_1^2 + b_1^2}$.

A well-designed PLL is used to track the operating frequency $f_1(t)$ of the voltage. Given the frequency and voltage data, the recursive LMS algorithm will be able to find the parameter a_1 and b_1 in (3.2), which allows the voltage magnitude A_1 to be calculated. A hypothesis test will then be used to determine whether a tap change occurs or not. The hypothesis test compares the feature of the voltage

fluctuation with that of tap changes to determine if the variation in voltage is caused by tap change or not.

In the following part of this chapter, the different sections of the tap change detection algorithm will be introduced in detail and the simulation results will be shown and discussed.

3.3 Frequency tracking technique

There are many different techniques that can be used for tracking the voltage frequency in power system. Zero crossing method is one of the most popular frequency measuring technique, which is easy to implement but cannot give accurate results when there is noise and high harmonic contents [106, 107]. Another method that has been widely used is the Discrete Fourier transform method [108–111]. It is easy to implement but limited by the length of the sliding data window as its output frequency resolution is limited. Phase-locked loops (PLL) have also been widely used. Compared to the zero crossing and Fourier transform methods, because the PLL uses a low-pass loop filter, it filters the noise and high harmonic contents in the input signal and thus provide a more accurate frequency estimate [112, 113]. Since the PLL has good dynamic performance, good precision of the frequency estimate and low calculation cost, the one used for our tap change detection algorithm is a PLL. There are many different types of PLLs. For example, the most basic one is the classical PLL. It is easy to design and has low calculation cost. Another one is the HTPLL, which has been discussed in Section 2.3.1. We will compare these two kinds of PLLs and pick the one that more suitable for our tap change detection algorithm.

3.3.1 Classical PLL

A classical PLL contains all three main components of PLL, namely PD, LF and VCO, and uses a multiplier as the PD, as described in Section 2.3.1. The input signal of a PLL can be expressed as $y_i = A_i \sin(2\pi f_i t + \theta_i)$, where the notation i signifies input. The output of the VCO is $y_o = A_o \cos(2\pi f_o t + \theta_o)$, where the notation o means output. In our algorithm, the input signal y_i is the periodic

signal obtained from power distribution system. Supposing that the input and output of a classical PLL are both pure sine wave signals and ignoring the noise and high harmonic contents, then for a multiplier-type PD, which compares the phase different between the input and output by multiplying them directly, we have

$$y_i \cdot y_o = \frac{1}{2} A_i A_o [\sin(4\pi f_i t + \theta_i + \theta_o) + \sin(\theta_i - \theta_o)]. \tag{3.3}$$

From (3.3), when a PLL is locked in, the high frequency component of its PD output is twice the frequency of the reference signal, while the low frequency content becomes a constant. When phase error $\Delta\phi = \theta_i - \theta_o$ is small, the sine function $y = \sin \Delta\phi$ can be approximated by a linear function $y \approx \Delta\phi$. A low-pass loop filter (LPLF) is needed by the PLL to filter the high frequency contents.

The design of the LPLF is very important for designing a PLL, which has been introduced in Section 2.3.1. Here we also adopt the method proposed in [96]. When using a PI controller as the LPLF, the LPLF has the following transfer function

$$C(s) = K\left(1 + \frac{1}{s\,T_i}\right). \tag{3.4}$$

Combining the three components of the PLL, the closed-loop transfer function is

$$G_e(s) = \frac{G_o(s)}{1 + G_o(s)} = \frac{\frac{KA}{2T_i} + s}{s^2 + \frac{KA}{2}s + \frac{KA}{2T_i}}, \tag{3.5}$$

where $A = \frac{1}{2} A_i A_o$. Comparing this with the standard second order transfer function, we can see that the natural frequency for this transfer function is

$$\omega_n = \sqrt{\frac{KA}{2\,T_i}}, \tag{3.6}$$

with damping factor

$$\zeta = \sqrt{\frac{KA\,T_i}{8}}. \tag{3.7}$$

In this system, ω_n and ζ are design parameters that are determined by the parameter K and T_i for the LPLF. In order to avoid overshoot in phase and achieve

good tracking, we choose $\zeta = 1$, so that,

$$K = \frac{4\,\omega_n}{A}, \tag{3.8}$$

$$T_i = \frac{2}{\omega_n}. \tag{3.9}$$

The natural frequency ω_n will affect the PLL's performance considerably: a large ω_n can provide faster reaction and better lock-in speed for the PLL against a step change, but on the other hand, this will also increase the amplitude of the high harmonic component in PD output and increase the over shoot in the transient process of the PLL output as well, which is shown in Figure 3.2. In Figure 3.2, the response of classical PLLs with same damping factor $\zeta = 1$ and different natural frequency from $\omega_n = 0.5\pi$ to $\omega_n = 4\pi$ with a step change in their inputs are compared. At $t = 5$s, the inputs of these PLLs change from 50Hz to 51Hz. It can be seen from the figure that when the natural frequency is smaller, the transient time for the PLL to react to the input step change is longer but the amplitude of the second order harmonic in its output is smaller. When $\omega_n = 0.5\pi$, the output frequency variation is $r_f = 1$Hz, when $\omega_n = \pi$, the output frequency variation is $r_f = 2$Hz, $\omega_n = 2\pi$, the output frequency variation is $r_f = 4$Hz and when $\omega_n = 4\pi$, the output frequency variation is $r_f = 8$Hz. In order to achieve a balance between reaction speed and reduce the presence of the higher harmonic in the PD output, we choose $\omega_n = 4\pi$.

In Figure 3.2, it worth to be mentioned that for the PLL with $\omega_n = 0.5\pi$, there shows an output fluctuation in the transient process. The reason for this phenomenon is that, when designing the PLL, we assume that the output and input frequency are very close to each other and the phase difference detected by the detector is very small. Only when such an assumption stands, can we assume that $\sin \Delta\phi \approx \Delta\phi$, and therefore the phase detection has linear behaviour (where $\Delta\phi = \theta_i - \theta_o$ is the phase difference between the input and output signal as mentioned in (3.3)). However when the difference between input frequency and output frequency are quite large, and the changing rate of the output is very slow, the phase difference $\Delta\phi$ can get larger and larger and the phase detector cannot

be regarded as linear device anymore. In addition, the control signal being sent to the VCO is $\sin \Delta\phi$ and $\sin x$ is a periodic function. If the output changing rate is vary slow, the phase difference $\Delta\phi = \theta_i - \theta_o$ will keep on increasing, and will exceed 2π. This will lead to the fluctuation of the VCO output frequency. The nonlinearity feature together with the period property of the phase detector lead to the output fluctuation in the transient process shown in Figure 3.2. When such phenomenon happens, the output frequency will not stop fluctuating until the PLL output frequency is getting close to the input frequency. In order to avoid such transient process, the parameter ω_n cannot be too small.

We can use MATLAB to simulate the properties of a classical PLL and test its response to different input frequency changes. The performance of a classical PLL is assessed by applying an abrupt change of the input signal frequency. The input signal is set to be a sine wave with a constant frequency of 50Hz for $0 \leq t < 5$, and the frequency is changed to 51Hz for $t \geq 5$. The output frequency of the classical PLL is as Figure 3.3 and Figure 3.4. The the red line is the input frequency, while the blue line is the output frequency of PLL.

Figure 3.3 shows the lock-in status of PLL. It can be seen from the figure that although the classical PLL has locked to the input signal, its output frequency is fluctuating around the input frequency. This is because even though we have included a low-pass filter, the second harmonic content in PD's output signal is not totally eliminated and the input signal of VCO will always contain high harmonic contents. Consequently, the output frequency of the PLL will be constantly fluctuating. Figure 3.4 shows the PLL's response to a step change of 1Hz in the input frequency. After the abrupt change in input frequency, the PLL adjusts its output frequency and locks in the input signal again in about 0.5 seconds and the overshoot is about 0.15Hz. It is worth mentioning that during the design process, we have made the classical PLL to be a second order system with damping factor $\zeta = 1$, so the output phase ϕ_o of the system should have no overshoot. However for the frequency f_o, there will be overshoot.

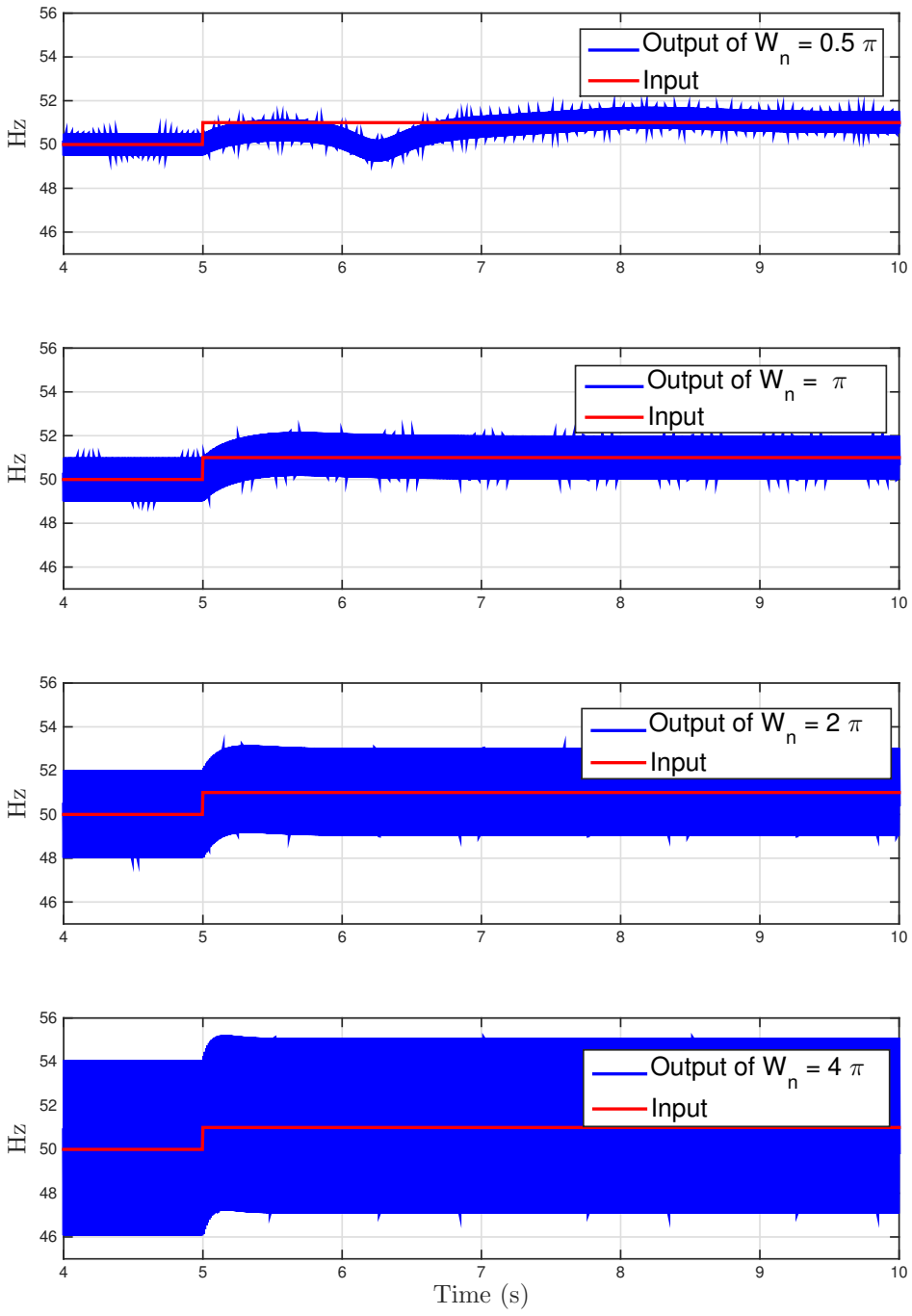


Figure 3.2: The responses of PLLs to a step change. The PLLs has the same damping factors of $\zeta = 1$ and different natural frequencies. Comparing the results, it can be seen that when the natural frequency is smaller, the transient time for the PLL to react to the input step change is longer but the amplitude of the second order harmonic in its output is smaller.

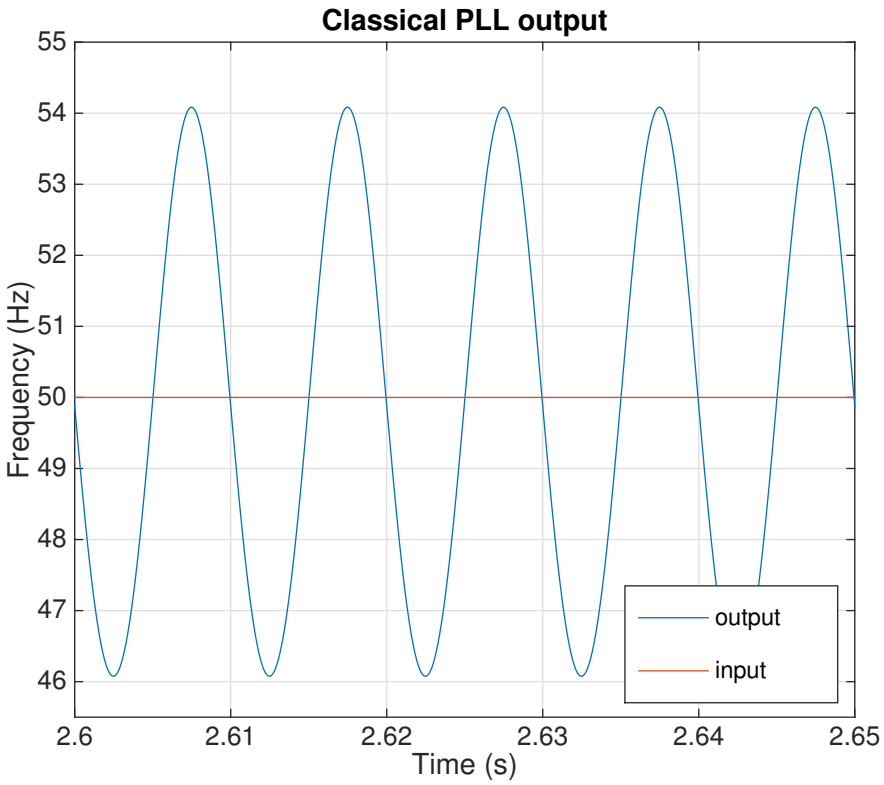


Figure 3.3: Classical PLL output in lock-in condition. The output is fluctuating around the input at twice the input frequency. This fluctuation is caused by the second order harmonic in the PD output.

Figure 3.5 shows the classical PLL's response to constant input frequency change, where the input signal frequency increases steadily. We see that when the input signal frequency is changing, the PLL tracks the input signal. Actually, since with a PI controller LPLF, the classical PLL is a Type 2 system, it can track the input frequency's change with no error when there is a linear (ramp) change in frequency.

In general, a classical PLL on its own can be used to track the frequency of the signal in power transmission system, however its tracking ability is not ideal, since its output frequency is constantly fluctuating rather than staying at the expected value due to the high harmonic content in its PD output.

3.3.2 Hilbert-transform PLL

The second order harmonic content in the output of classical PD can lead to the fluctuation in the PLL's output frequency. One of the solution to this problem is

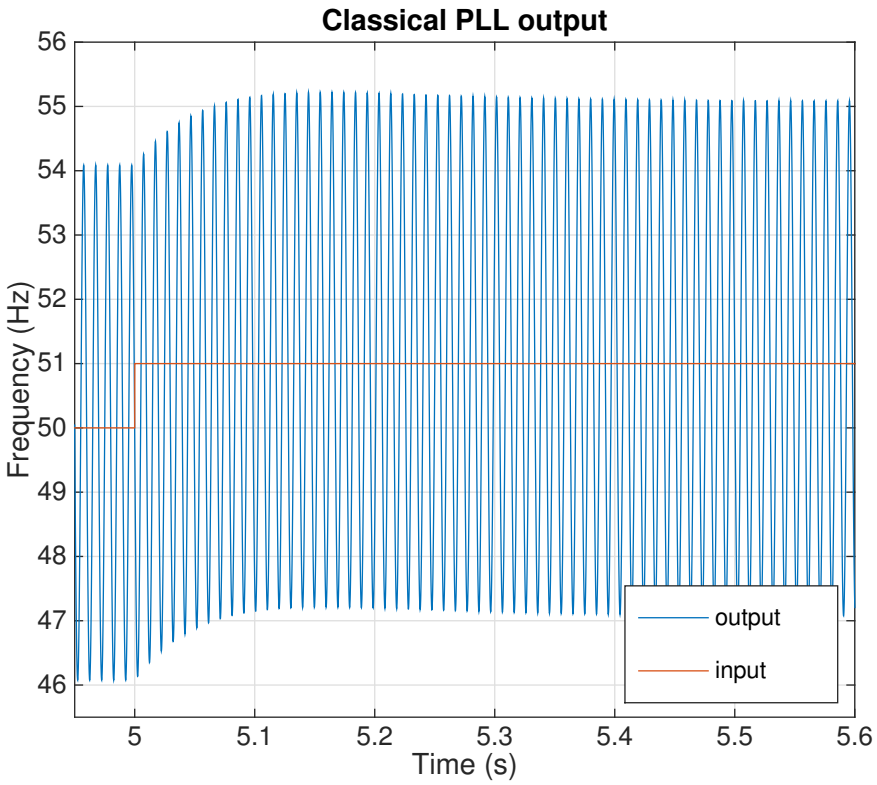


Figure 3.4: Classical PLL response to input frequency change. The input step change occurs at $t = 5$s. The PLL output shows a transient process and locked-in to the input in about 0.5s.

to substitute the multiplier PD with a Hilbert-transform PD (HTPD) [97], which will make the PLL a Hilbert-transform PLL (HTPLL)

An HTPLL has all of the three main sections of a PLL, so a low pass loop-filter and a VCO are also required. Following the LPLF designing method proposed in [96], our HTPLL is set to have the same PI controller as the classical PLL and is tested with the same input signal mentioned in the previous simulation. Figure 3.6 and Figure 3.7 show the output of HTPLL. From these figures, we can see that comparing to the performance of the classical PLL, the output frequency of HTPLL is not affected by the second harmonic. The frequency estimation result of HTPLL is much clearer than that made by classical PLL. Even though the HTPLL's output frequency is also fluctuating slightly around the input frequency when there is noise in the input, compared to the output from the classical PLL, this small fluctuation is negligible. In Figure 3.6, when the input frequency is abruptly changed, the

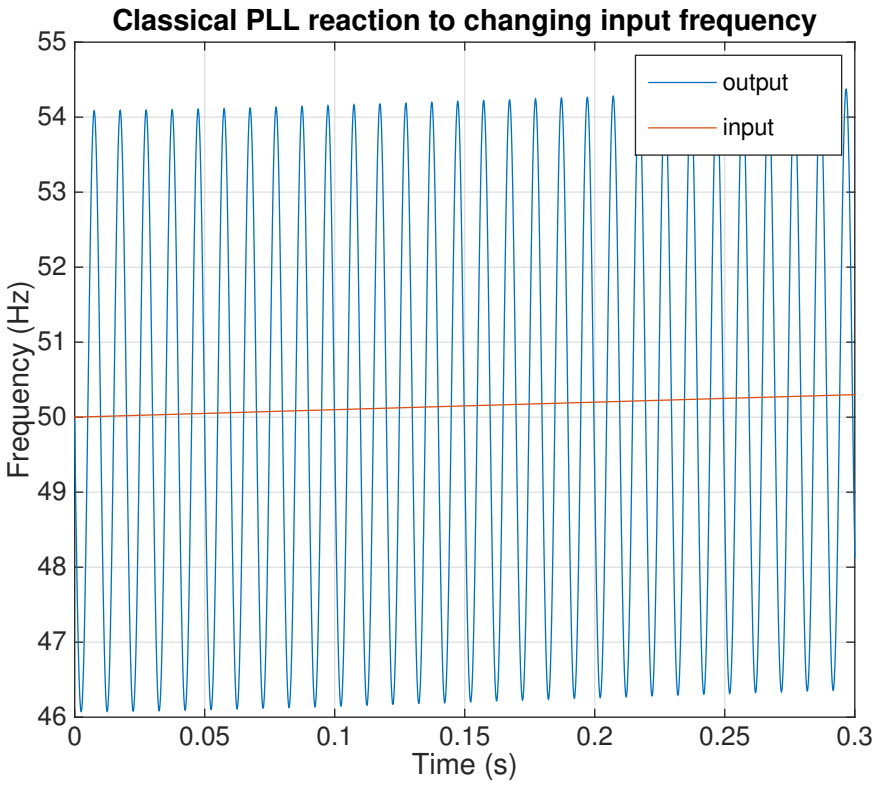


Figure 3.5: Classical PLL response to changing input frequency

HTPLL adjusts itself and locks in at the input frequency in about 0.5 seconds with an overshoot about 0.1Hz, which is very similar to the performance of traditional PLL. When the input frequency follows a sawtooth curve, the HTPLL can still track the input signal. It can be seen in Figure 3.7 that for the filter window introduced by Hilbert transformer, there will be a delay between the HTPLL output and input frequency. Also, when the voltage frequency changes from increasing to decreasing, there will be a transient response in the output. When the frequency changes smoothly towards one direction, as the HTPLL is a Type 2 system, ignoring the delay caused by HTPD window, it can track slope frequency variations with no steady state error. It worth mentioning that there is a small gap between the input and PLL output frequency in Figure 3.7, but this gap is not caused by the steady state error, but is due to the time delay introduced by the Hilbert-tansform.

Compared to the classical PLL, the HTPLL's output frequency is more stable and its frequency tracking ability is good as well, so the HTPLL is used as a

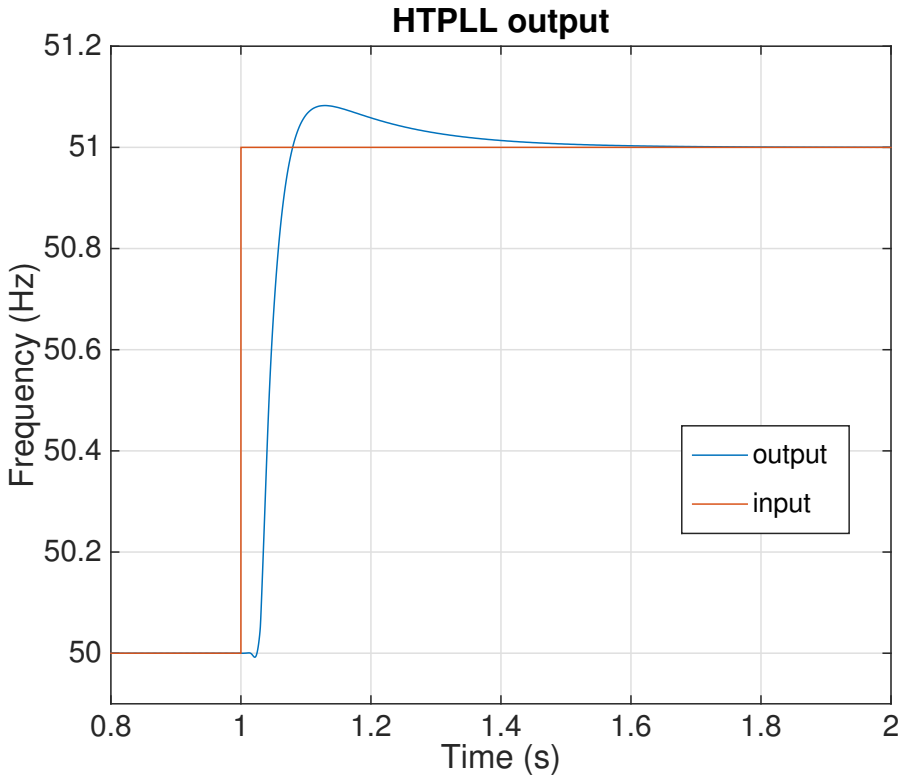


Figure 3.6: HTPLL response to a step input frequency change. As the HTPLL uses a HTPD which dose not generate a second order harmonic, output of HTPLL is clearer than that of the classical PLL

frequency tracking method for the tap change detection algorithm.

3.4 Recursive LMS algorithm

In order to find the amplitude of the input sine wave signal, we need not only the frequency tracking section, but also a linear regression algorithm that can find a_i and b_i in (3.2), so that magnitude can be obtained. In the real data obtained from a power system, in addition to the 50Hz sinusoidal data, there are also high frequency harmonics, mainly due to the transformers and the loads in the power transmission system. A tap change is detected as a change in the amplitude of the fundamental frequency signal. Thus, we only need to find the amplitude of the first harmonic of the signal.

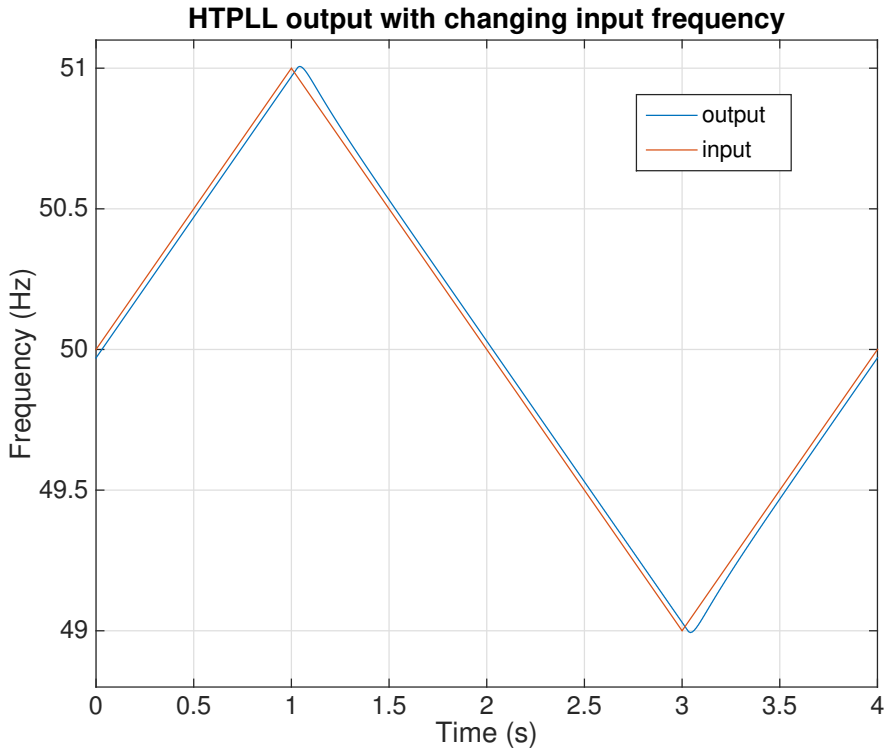


Figure 3.7: HTPLL reaction to input frequency fluctuation

3.4.1 Classical recursive least square algorithm

Given a sampled voltage signal y_i, we can use the HTPLL to find the phase for each data point, so that the magnitude of the given set of signal data can be found using the linear regression algorithm. If the data is denoted $y_1, y_2, y_3, ..., y_d$ and $x_{1,1}, x_{1,2}, ..., x_{1,n}, x_{2,1}, ..., x_{d,n}$ (where $x_{i,k} = sin(2\pi f t_k)$), the linear regression algorithm finds the linear relationship between y_i and $x_{i,1}, \cdots, x_{i,n}$

$$y_i = \beta_1 x_{i,1} + \beta_2 x_{i,2} + ... + \beta_d x_{i,d} + \varepsilon_i, \tag{3.10}$$

where β_1 to β_d are the parameters of the model and ε_i are error terms. This can also be expressed in matrix format as

$$Y = X\beta + \epsilon, \tag{3.11}$$

where

$$\begin{aligned} Y &= [y_1, y_2, y_3, ..., y_d]^T, \\ \beta &= [\beta_1, \beta_2, ..., \beta_d]^T, \end{aligned} \tag{3.12}$$

and

$$X = \begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_d^T \end{bmatrix} = \begin{bmatrix} x_{1,1}x_{1,2}\cdots x_{1,n} \\ x_{2,1}x_{2,2}\cdots x_{2,n} \\ \vdots \quad \ddots \vdots \\ x_{d,1}x_{d,2}\cdots x_{d,n} \end{bmatrix}, \tag{3.13}$$

In least squares estimation, the sum of the square of error terms,

$$V(\beta) = \sum \varepsilon_i^2 = \epsilon^T \epsilon = (Y - X\beta)^T (Y - X\beta). \tag{3.14}$$

is minimised, which occurs when [114]

$$\beta = (X^T X)^{-1} (X^T Y) = \left(\sum x_i \, x_i^T\right)^{-1} \left(\sum x_i \, y_i\right), \tag{3.15}$$

The expression in (3.15) can be used to calculate the linear regression amplitude result based on least square estimate for any given set of voltage data and the corresponding phase value.

The linear regression method described above is an off-line method, but in our problem, we need to calculate the result every time when we obtain a new data. In order to improve the calculating efficiency and thus make our algorithm be capable of making on-line calculation, we need to use an adjusted version of the LMS algorithm which is the Recursive Least Square (RLS) algorithm [85, 115].

From (3.15), we can define

$$P(t) = \left(\sum_{i=1}^{t} x_i \, x_i^T\right)^{-1}, \tag{3.16}$$

then

$$P^{-1}(t) = P^{-1}(t-1) + x_t \, x_t^T. \tag{3.17}$$

which leads to the recursive least square algorithm [114]

$$\beta(t) = \beta(t-1) + K(t)\,\varepsilon(t), \tag{3.18}$$

$$K(t) = P(t)\,x_t, \tag{3.19}$$

$$\varepsilon(t) = y_t - x_t^T \beta(t-1), \tag{3.20}$$

where the equation for updating $P(t)$ and $K(t)$ are as follows

$$P(t) = P(t-1) - \frac{P(t-1)\,x_t\,x_t^T P(t-1)}{1 + x_t^T P(t-1)\,x_t}, \tag{3.21}$$

$$K(t) = \frac{P(t-1)\,x_t}{1 + x_t^T P(t-1)\,x_t}. \tag{3.22}$$

We can use Matlab to implement the RLS algorithm. A set of input voltage data with the frequency of $f = 50$Hz and sampling frequency of $f_s = 10^4$Hz are created, where the amplitude of the signal is $A_1 = 230\sqrt{2}$ when $0 \leq t < 4$ and changes to $A_2 = 240\sqrt{2}$ for $t \geq 4$. We use the input voltage signal $V_a(t) = a_i \cos(2\pi\, f\, t) + b_i \sin(2\pi\, f\, t)$, where $a_i = A_i\, \cos\theta$, and $b_i = A_i\, \sin\theta$, where θ is the phase difference of PLL output and the input signal. Then, with the phase $\varphi(0) = 0$, $\varphi(t) = \int_0^t 2\pi f t\, dt$ and voltage value V_a we can obtain a_i and b_i with RLS algorithm. The amplitude of the signal can be calculated with the following equation

$$A_i = \sqrt{a_i^2 + b_i^2}. \tag{3.23}$$

The simulation result of RLS algorithm is shown in Figure 3.8, where the blue line shows the input signal amplitude and the red line is the amplitude calculated using RLS algorithm. We can see that when the input amplitude changes, the RLS algorithm estimate will change towards the new input magnitude, but will take time for the algorithm to converge to the correct value. The transient process for the algorithm takes a long time since the algorithm still includes the old data into calculation. In order to make the algorithm react quickly to input magnitude changes and thus be more sensitive to the variation in the input signal, the algorithm should be capable of forgetting the information of old data. One approach is to use a sliding window in the algorithm and the Windowed Recursive Least Square (WRLS) algorithm is an on-line algorithm that calculates the least square estimation for data in a sliding window.

It is straightforward to change the least square estimation (LSE) to a WRLS algorithm. From (3.15), we can see that if we can save the following two parameter $\sum x_i\, x_i^T$ and $\sum x_i\, y_i$, we will be able to update and downdate the data window,

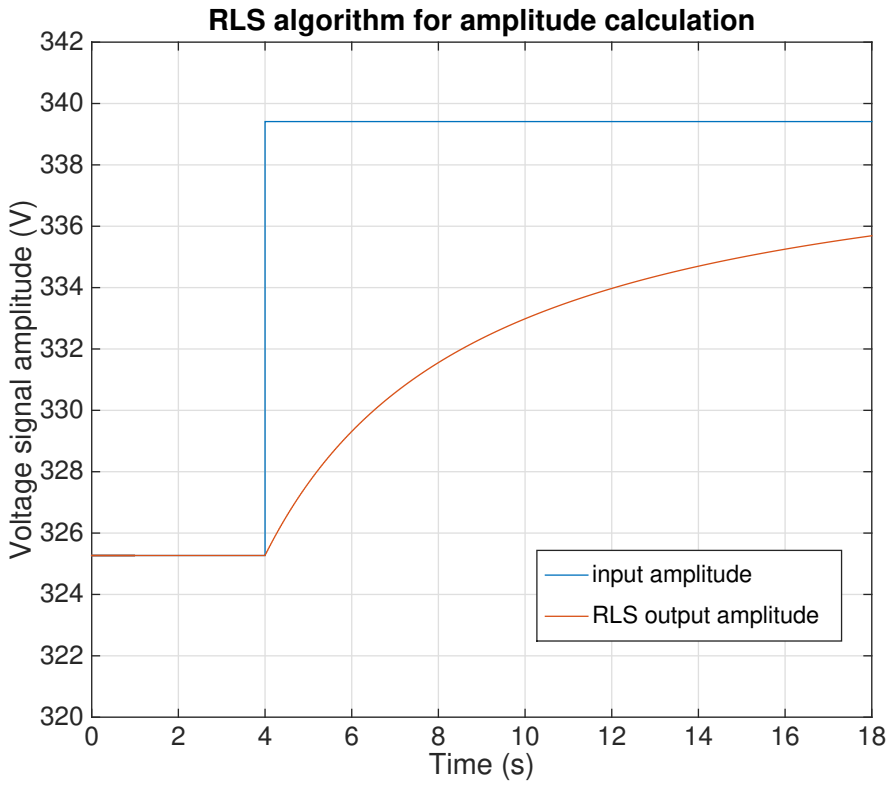


Figure 3.8: RLS algorithm result for amplitude calculation. As the RLS algorithm takes all of the data into calculation, it reacts to input step change slowly.

which means the output value can be updated every time when a new data is put into the algorithm. In such a way, the classical LSE can be changed to an on-line WRLS algorithm.

If the independent variables vector is x_i, the dependent variable is y_i, we can define that

$$\begin{aligned} P_w(t) &= \left(\sum_{i=t-l+1}^{t} x_i x_i^T \right)^{-1}, \\ S_w(t) &= \sum_{i=t-l+1}^{t} x_i y_i, \end{aligned} \tag{3.24}$$

where l is the length of the sliding window. Defining $\beta(t)$ as the least square estimation based on $x_{t-l+1}, \cdots, x_t$ and $y_{t-l+1}, \cdots, y_t$ and comparing (3.15) with (3.24), gives

$$\beta(t) = P_w(t)\, S_w(t). \tag{3.25}$$

In the WRLS algorithm, $P_w(t)$ and $S_w(t)$ are saved so that the update and downdate process can be achieved. When the next set of data x_{t+1} and y_{t+1} arrives, the result can be updated using

$$\widehat{P_w}(t+1) = \left(\sum_{i=t-l+1}^{t} x_i x_i^T + x_{t+1} x_{t+1}^T \right)^{-1} = [P_w(t)^{-1} + x_{t+1} x_{t+1}^T]^{-1}, \tag{3.26}$$

and

$$\widehat{S_w}(t+1) = \sum_{i=t-l+1}^{t} x_i y_i + x_{t+1} y_{t+1} = S_w(t) + x_{t+1} y_{t+1}. \tag{3.27}$$

$\widehat{P_w}(t+1)$ and $\widehat{S_w}(t+1)$ use both the data in previous window and the new data x_{t+1}, y_{t+1}. In order to keep the length of the window to be l, the old data x_{t-l+1} and y_{t-l+1} are forgotten, which is the downdate step. The downdate process can be calculated by the following equations

$$P_w(t+1) = \left(\sum_{i=t-l+2}^{t+1} x_i x_i^T \right)^{-1} = [\widehat{P_w}(t+1)^{-1} - x_{t-l+1} x_{t-l+1}^T]^{-1}, \tag{3.28}$$

similarly,

$$S_w(t+1) = \sum_{i=t-l+2}^{t+1} x_i y_i = \widehat{S_w}(t+1) - x_{t-l+1} y_{t-l+1}. \tag{3.29}$$

With the updated and downdated equations, the least square estimation result for the next window $\beta(t+1)$ can be calculated as

$$\beta(t+1) = P_w(t+1)^{-1} S_w(t+1). \tag{3.30}$$

It should be mentioned that while choosing the length of the data window, there is a trade off between the accuracy and the reaction speed. A longer data window will provide the algorithm with better accuracy as the estimate is based on more information, but when there is a change in the input data, the algorithm will react more slowly. For our algorithm, we set the length of the window to be $l = 600$, which, approximately corresponds to the length of three whole periods for the input signal. Using this window length, the WRLS can achieve both good reaction speed and good accuracy. We test the performance of the WRLS algorithm using the same simulated input data used in the RLS algorithm simulation.

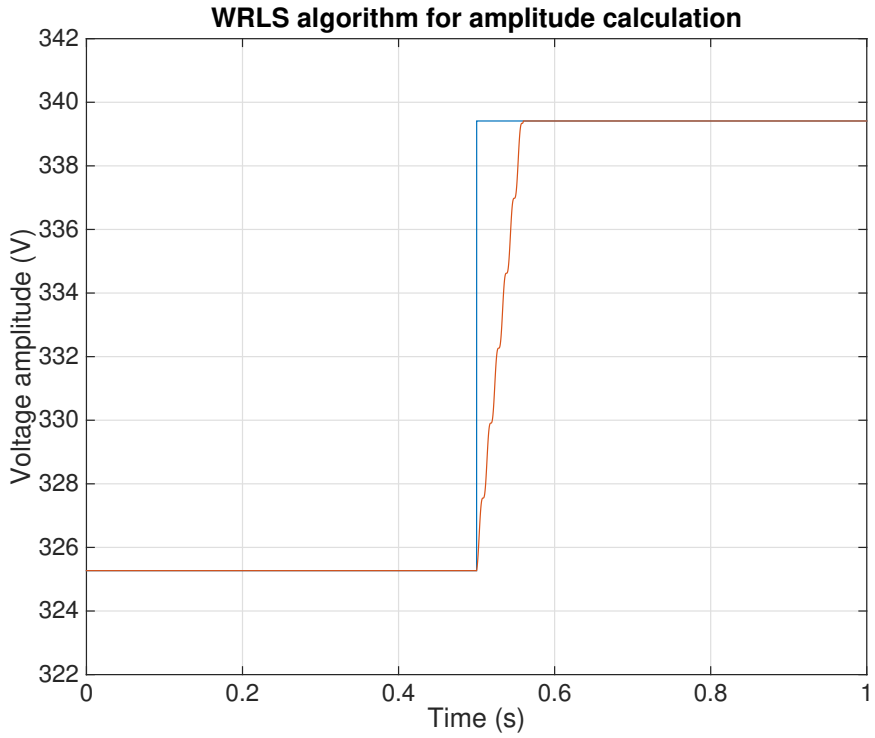


Figure 3.9: WRLS algorithm result for amplitude calculation. There is a step change in the input magnitude at $t = 0.5$s. After the change occurs, the WRLS algorithm reacts to the step change quickly.

Figure 3.9 shows the simulation result of WRLS algorithm, where the blue line is the input signal amplitude and the red line is the calculation result of the algorithm. The figure shows that following the change in the input signal amplitude, the WRLS algorithm estimate converges to the new value after about 0.06 seconds. As the sampling frequency of the data is $f_s = 10^4$Hz, the algorithm requires $t = l/f_s = 0.06s$ to forget the data with the old amplitude and respond to data with the new amplitude.

An interesting pattern in Figure 3.9 is that the output of the WRLS algorithm is not moving towards the expected value smoothly but bouncing while increasing towards the input amplitude. This phenomenon is caused by the mismatch between the period of the signal and the length of the sliding window. As we set the length of the window of the WRLS algorithm to be $l = 600$, which theoretically should contain three full cycles. Such an assumption is true when the frequency of the

power system is exactly 50Hz and the sampling frequency of the input is 10kHz. However, in reality, it is very difficult to promise that the sliding contains exactly three full cycles of the sinusoidal data. The mismatch between the length of the sliding window and the cycles of data will lead to the error in the WRLS algorithm output. That is the reason why the curve shows a slightly fluctuating when the output is moving towards the changed input amplitude.

3.4.2 QR decomposition windowed recursive least square algorithm

Even though the classical windowed recursive least square algorithm can be used to find the amplitude of the voltage, it has a major disadvantage. In WRLS algorithm, the inverse matrix of covariance matrix $P_w(t)$ need to be updated at every time step, which means the inverse matrix has to be calculated each time, and when there occur $P_w(t)$ is not positive definite or $P_w(t)$ is not symmetric, this method may not give the correct result. In real implementations, this may lead to unpredictable problems [116–119]. Therefore, in order to achieve numerical robustness and void calculating inverse matrix each time, we adopt the QR Decomposition Windowed Recursive Least Square (QRD-WRLS) algorithm [120, 121] and in this project, the downdating algorithm proposed by Yoo and Park is used [122]. This algorithm can achieve good accuracy and avoid the problems associated with the classical WRLS method.

For any given least square estimate in (3.11), we apply a QR decomposition to the data matrix X,

$$X = QR, \tag{3.31}$$

where $X \in \mathbb{R}^{d\times n}$ is the data matrix containing d independent variable vectors, with $d > n$, $Q \in \mathbb{R}^{n\times n}$ is a orthogonal matrix and $R \in \mathbb{R}^{d\times n}$ is an upper triangular matrix. We can then have

$$QX = \begin{pmatrix} R_1 \\ 0 \end{pmatrix}, \quad QY = \begin{pmatrix} u \\ v \end{pmatrix}, \tag{3.32}$$

where $R_1 \in \mathbb{R}^{n\times n}$ is a square, upper triangular matrix, $u \in \mathbb{R}^{n\times 1}$ and $v \in \mathbb{R}^{(d-n)\times 1}$. (3.14) can be rewritten as

$$\|\epsilon\|^2 = \|(Y - X\beta)\|^2 = \|Q(Y - X\beta)\|^2. \tag{3.33}$$

Combining (3.32) and (3.33),

$$V(\beta) = \|\epsilon\|^2 = \|R_1\beta - u\|^2 + \|v\|^2, \tag{3.34}$$

so $V(\beta)$ is minimized when

$$R_1\beta = u. \tag{3.35}$$

For QRD-WRLS, the relation between the data matrix at time step $t = k$ and time $t = k + 1$ can be illustrated as the following sequence

$$X(k) \xrightarrow{update} \begin{pmatrix} X(k) \\ x_{d+1}^T(k+1) \end{pmatrix} \rightarrow \widetilde{X}(k+1) \rightarrow \begin{pmatrix} X(k+1) \\ x_{d+1}^T(k+1-d) \end{pmatrix} \xrightarrow{downdate} X(k+1)$$

here $X(k), X(k+1) \in \mathbb{R}^{d\times n}$ are data matrix containing information of all the data in the sliding window with the length of d, $x(i)$ is data vector following $x(i) \in \mathbb{R}^{n\times 1}$, $\widetilde{X}(k+1)$ is the data matrix after updated, which follows $\widetilde{X}(k+1) \in \mathbb{R}^{(d+1)\times n}$. The data matrix is first updated to include the information from the new data point $x(k+1)$ and then downdated to delete the information of the old data $x(k+1-d)$ to recover the data matrix to the downdated result $X(k+1)$. However, it is the upper triangular matrix R_1 and vector u that being updated at each time step.

For the update step, the following equation is used

$$\widehat{Q}^T(k+1)\begin{pmatrix} R_1(k) \\ 0 \\ x_{d+1}^T(k) \end{pmatrix} = \begin{pmatrix} \widetilde{R}_1(k) \\ 0 \\ 0 \end{pmatrix}, \tag{3.36}$$

where $\widehat{Q}(k+1) \in \mathbb{R}^{d+1\times d+1}$ is the updating orthogonal matrix and $\widetilde{R}_1(k) \in \mathbb{R}^{n\times n}$ is the updated triangular matrix containing the information of $d+1$ data vector from $x_1(k)$ to $x_{d+1}(k)$. $\widehat{Q}(k+1)$ has the following format

$$\widehat{Q}(k+1) = \begin{pmatrix} A(k+1) & 0 & q(k+1) \\ 0 & I_{d-n} & 0 \\ p^T(k+1) & 0 & \lambda(k+1) \end{pmatrix}, \tag{3.37}$$

where $A(k+1) \in \mathbb{R}^{n\times n}$ and $p(k+1), q(k+1) \in \mathbb{R}^{n\times 1}$, then the updated $\widetilde{u}(k+1)$ can be obtained with the following equation,

$$\widetilde{u}(k+1) = A^T(k+1)u(k) + p(k+1)y_{d+1}(k+1). \tag{3.38}$$

In order to move the sliding window forward, the old data needs to be deleted with the downdate step, which can be achieved by removing the old data $x_1^T(k+1)$ using

$$\check{Q}^T(k+1)Q^T \begin{pmatrix} x_1^T(k+1) \\ X(k+1) \end{pmatrix} = \check{Q}^T(k+1) \begin{pmatrix} \widetilde{R}_1(k+1) \\ 0 \end{pmatrix} = \begin{pmatrix} R_1(k+1) \\ 0 \\ x_1^T(k+1) \end{pmatrix}. \tag{3.39}$$

From (3.39) we can see that $\check{Q}^T(k+1)Q^T$ has the following structure

$$\check{Q}(k+1)Q = \begin{pmatrix} 0_{1\times d} & 1 \\ \widetilde{Q}(k+1) & 0_{d\times 1} \end{pmatrix}. \tag{3.40}$$

and, $\check{Q}^T(k+1)$ can be recovered with a Givens rotation [123]. Supposing $\check{Q}^T(k+1)$ can be expressed as

$$\check{Q}(k+1) = J_{d,d+1}Jd-1, d+1, \cdots, J_{1,d+1}. \tag{3.41}$$

we can express $\check{Q}^T(k+1)$ as the product of two matrix

$$\check{Q}(k+1) \triangleq \check{Q}(k+1)_t\check{Q}(k+1)_b, \tag{3.42}$$

where

$$\check{Q}(k+1)_b = J_{d,d+1}J_{d-1,d+1}, \cdots, J_{n+1,d+1}, \quad \check{Q}(k+1)_t = J_{n,d+1}J_{n-1,d+1}, \cdots, J_{1,d+1}. \tag{3.43}$$

Then, matrix $\check{Q}(k+1)_t$ can be expressed as

$$\check{Q}(k+1)_t = \begin{pmatrix} B(k+1) & 0 & g(k+1) \\ 0 & I_{d-n} & 0 \\ f^T(k+1) & 0 & \mu(k+1) \end{pmatrix}, \tag{3.44}$$

where $B(k+1) \in \mathbb{R}^{n\times n}$ and $f(k+1), g(k+1) \in \mathbb{R}^{n\times 1}$

So, the vector $u(k+1)$ can be obtained from the following equation,

$$u(k+1) = B^T(k+1)\widetilde{u}(k+1) + f(k+1)v'_{d-n+1}(k+1), \tag{3.45}$$

where $v'_{d-n+1}(k+1)$ can be obtained from

$$y_1(k+1) = g^T(k+1)\widetilde{u}(k+1) + \mu(k+1)v'_{d-n+1}(k+1). \tag{3.46}$$

In order to test the working properties of the QRD-WRLS algorithm, we simulate the QRD-WRLS algorithm using Matlab and run the same input data as for the RLS algorithm test. The length of the window is set to be $l = 600$, which is the same as the previous trials. The test result is shown in Figure (3.10) From the

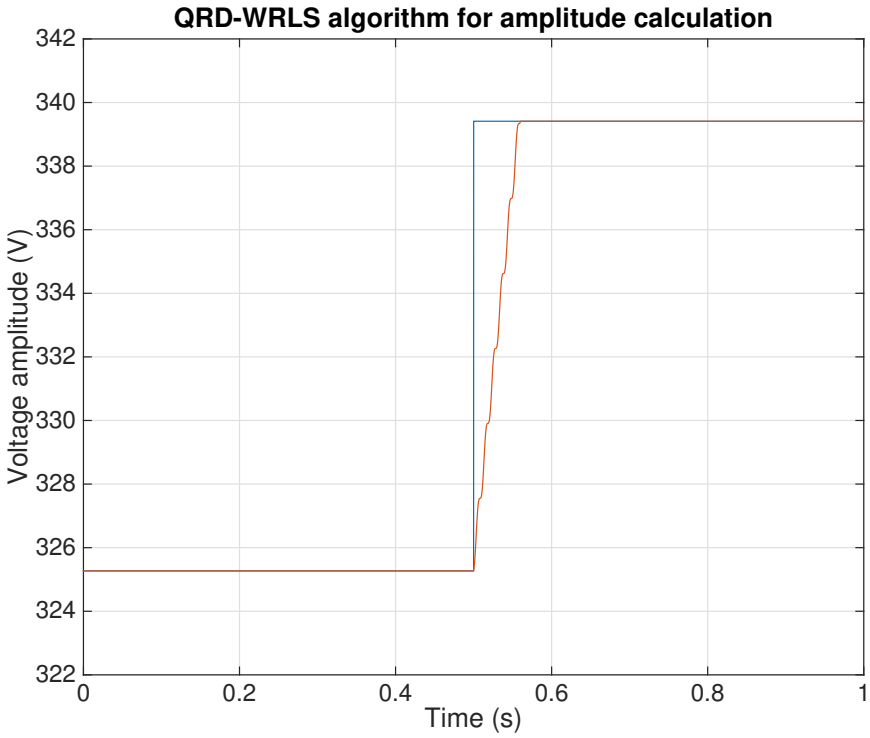


Figure 3.10: QRD-WRLS algorithm result for amplitude calculation

figure, we can see that the QRD-WRLS algorithm has the same reaction speed and accuracy as classical WRLS algorithm. However, QRD-WRLS is numerically robust which means it can provide the more reliable results compared to the normal WRLS method. In the figure, the output also shows a slightly fluctuating pattern when moving towards the changed input. The reason for this phenomenon is the same as described for that of Figure 3.9.

3.5 Hypothesis test

With a well designed HTPLL and the WRLS algorithm, given the sinusoidal voltage input data, the magnitude of the input signal can be estimated. In order to detect tap changes which cause changes in the voltage magnitude variation, we also need a method that can test the voltage variation and distinguish tap changes from other kind of fluctuations. As previously mentioned, a tap change is controlled by on-load tap changer (OLTC) transformer, which adjusts the output transformer ratio, changing its output voltage. A typical tap change is shown in Figure 3.11 (only the top of the sinusoidal voltage signal is shown-note the scale on the voltage axis). The tap change can be detected by comparing the amplitude of the voltage before and after a tap change. We use a hypothesis test in our algorithm to detect the amplitude variation and thus know whether a change in voltage is due to a tap change.

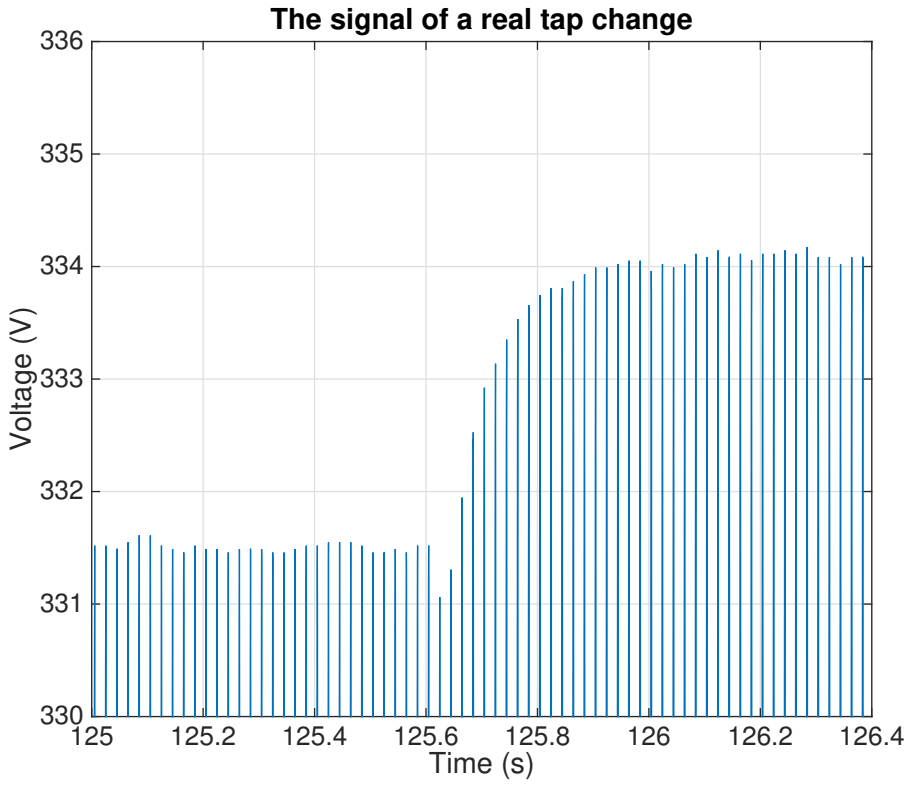


Figure 3.11: Signal of a real tap change (only the top portion of the sinusoidal voltage signal is shown)

3.5.1 Generalized likelihood ratio test

There are a number of statistical tests that can be used to detect changes from a set of fluctuating signal with noise [124–126], and a commonly used method is the generalized likelihood ratio test. Likelihood is a measure of likeliness of what has been observed when given a certain assumption and observed data. Using the likelihood result, different assumptions on the time at which a change in amplitude occurred can be compared [127]. For a set of input data model following normal distribution such as

$$y_t = \theta + e_t, \quad e_t \in N(0, R), \tag{3.47}$$

where y_t is the input data, θ is the mean value and e_t is the Gaussian noise, the likelihood is given by the following function,

$$l_t(\theta, R) \triangleq p(y^t|\theta, R) = \prod_{i=1}^{t} p(y_i|\theta, R) = (2\pi R)^{-t/2} e^{1/(2R)\sum_{i=1}^{t}(y_i-\theta)^2}, \tag{3.48}$$

where, $l_t(\theta, R)$ is the likelihood for the set of data $y^t = \{y_1, \cdots, y_t\}$ given the distribution parameters of mean θ and variance R. For change detection, however, what we desire is the change time, so θ and R are redundant parameters that can be eliminated by Maximum Likelihood (ML) estimation [128].

The change time can be estimated by the maximum likelihood principle by the following equation,

$$\hat{k} = \arg\max_k p(y^t|k), \tag{3.49}$$

where $p(y^t|k)$ is the likelihood for measurements $y^t = y_1, y_2, ..., y_t$. Based on generalized likelihood estimation, supposing that the distribution of data follows a Gaussian distribution, we have the following equation when θ and the noise variance R is known [129]

$$-2\log[l_t^{MGL}(T)] \approx t\log(2\pi R) + \frac{t\hat{R}}{R}, \tag{3.50}$$

where $\hat{R}$ is the actual variance of the data set $y_1, y_2, ..., y_t$. If we assume that before and after the change, the values of θ are independent, then the likelihood can be divided into two parts,

$$p(y^t|k) = p(y^k|k)p(y_{k+1}^t|k) = p(y^k)p(y_{k+1}^t), \tag{3.51}$$

and we can use the likelihood ratio to compare the likelihood of the null hypothesis, against the change hypothesis.

$$H_0 : \text{no change happens}$$
$$H_1(k, \nu) : \text{a jump of magnitude } \nu \text{ at time } k$$

The log likelihood ratio for the hypothesis test is

$$g_t(k, \nu) \triangleq 2 \log \frac{p(y^t|H_1(k, \nu))}{p(y^t|k = t)}, \tag{3.52}$$

using (3.51) and (3.52), we can calculate the likelihood ratio for the hypothesis test.

The classical generalized likelihood ratio method is an offline algorithm that maximizes both of the change time k and the change amount ν. For the problem we are considering, the amplitude change ν may vary in a certain range, and the change can happen at any time, so for tap change detection we need an on-line approach.

In order to adapt the classical off-line GLR method to an on-line process and decrease the calculation needed for the test, we use a sliding window and calculated the GLR result within the window every time the data in the window is updated. The following sequences (3.53) shows how the algorithm is working with a moving window.

$$\begin{aligned}
&y_1, y_2, \cdots y_{t-2d}, \underbrace{\overbrace{y_{t-2d+1}, y_{t-2d+2}, \cdots, y_{t-d}}^{W_1}, \overbrace{y_{t-d+1}, y_{t-d+2}, \cdots, y_t}^{W_2}}_{W_l}, y_{t+1}, y_{t+2}, \cdots \quad (a)\\
&y_1, y_2, \cdots y_{t-2d}, y_{t-2d+1}, \underbrace{\overbrace{y_{t-2d+2}, \cdots, y_{t-d}, y_{t-d+1}}^{W_1}, \overbrace{y_{t-d+2}, \cdots, y_t, y_{t+1}}^{W_2}}_{W_l}, y_{t+2}, \cdots \quad (b)
\end{aligned} \tag{3.53}$$

A large sliding window W_l with the length of $2d$ is divided equally into two smaller windows, W_1 and W_2 containing equal number of data d, as shown in (3.53)(a). Then the GLR result for the data in W_l can be calculated using

$$g_t(W_l) = \frac{l_{W_1}^{MGL} \cdot l_{W_2}^{MGL}}{l_{W_l}^{MGL}}. \tag{3.54}$$

The calculation of (3.54) involves (3.50), which requires the variance $\hat{R}$. It can be calculated by $\hat{R} = \bar{y^2} - \bar{y}^2$. It is straightforward to update and downdate the sum and quadratic sum of the data in these windows. Every time step, a new data is received and the sliding window moves forward, as shown in (3.53)(b), and the parameters of these sliding windows need to be updated and downdated. The GLR test for the sliding window indicates the probability that a change in the input data happens at the mid-point of the large sliding window.

The result of this GLR test is a likelihood ratio, but some information about the change needs to be known in advance so that the GLR test threshold can be calculated. The threshold $h(\nu)$ for any abrupt change with change amount ν can be calculated by assuming the data change from $y_i = p$ to $y_{i+1} = p + \nu$. With (3.50), (3.51) and the analysis above, we can have the following equation for threshold calculation [129, 130]

$$h(\nu) = \frac{l\hat{R}_l}{R_l} - \frac{l_1\hat{R}_1}{R_1} - \frac{l_2\hat{R}_2}{R_2}. \tag{3.55}$$

where R_l, R_1, R_2 are the Gaussian distribution variance of the data, and $\hat{R}_l$, $\hat{R}_1$, $\hat{R}_2$ are the variance calculated from the data set.

The GLR test statistic g_t^G follows the χ^2 distribution [124, 131], so the distribution of g_t can be obtained using (3.52). The tap change detection algorithm is designed to operate with a large amount of data, so in order to achieve high reliability, the confidence interval for the GLR test threshold h_g need to be higher than 99.99%. The test threshold for the log generalized likelihood ratio is $h_L = 2 \log h_g$, where h_g is the generalized likelihood ratio test threshold. An improved threshold can be obtained by considering the feature of tap change outlined in Fig. 3.11. If a tap change leads to a V_c voltage change in T_c seconds, the length d of the moving

data windows W_1, W_2 are small enough, and data is sampled with the sampling frequency of f_s Hz. We assume that the voltage is increasing linearly and there is no noise during the tap change, the GLR value of the tap change can be calculated with (3.54), and it can be rearranged as

$$g_{t,min} = \frac{V_c^2 d^2 (3d-2)}{6T_c^2 f_s^2 R}. \tag{3.56}$$

From (3.50) it can be concluded that when there is a voltage fluctuation, the variance of the data will be larger. Therefore the test statistic will be larger than $g_{t,min}$, so $g_{t,min}$ can be used as the test threshold as long as it can provide the confidence interval to be higher than 99.99%. In practice, compared to the variance of the voltage magnitude R, the voltage change V_c during tap change is very large and hence the confidence interval requirement can be easily satisfied. For example, for real data, variance of the noise is $R = 0.3\text{V}^2$, which is obtained from the test on real data, the threshold h_L for 99.99% is 12.8, whereas $g_{t,min}$ is around 280. It is reasonable to use this $g_{t,min}$ as the test threshold because the GLR result will always be larger when the variance of the data $\hat{R}_l$, $\hat{R}_1$ and $\hat{R}_2$ is bigger, and in (3.55) we assume that $\hat{R}_l = \hat{R}_1 = \hat{R}_2 = 0$, so the threshold $h(\nu)$ is the minimum GLR value that a change can be.

In the test on real-time data, due to the noise and fluctuation in the data, the GLR test result for a tap change is larger than $h(\nu)$. However the GLR statistic can also reach the threshold even if there is no change due to the variability in the signal

The properties of the on-line GLR method were tested with a set of simulated data that consists of a step change in the mean at $t = 1$s

$$y_t = \begin{cases} 0 + e(t), & for\ 0 < t \leq 1, \\ 1 + e(t), & for\ 1 < t \leq 2, \end{cases}$$

where $e(t)$ is a Gaussian distribution noise which follows $e(t) \in N(0,1)$, the data is sampled with sampling frequency of $f_s = 10^4$Hz. The data window was chosen to be $l_1 = l_2 = 200$.

Figure 3.12 shows the result of using the GLR test on the simulated data, from which we can see that the GLR statistic is fluctuating but is always smaller than

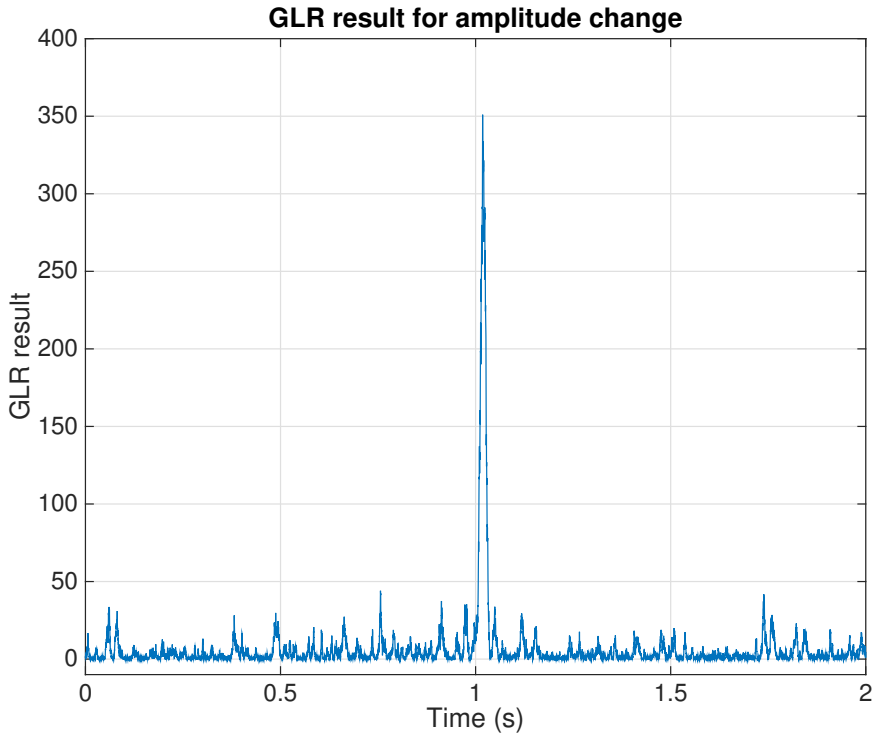


Figure 3.12: GLR result for change detection. When the change occurs at $t = 1$s, the GLR statistics increases and exceeds the threshold quickly, whereas when there is no change, the result is comparatively small.

the threshold which is $g_{t,min} = 280$ when there is no change. However, when the change occurs, the GLR statistic exceeds the threshold. The highest value of GLR occurs when the two data windows are at the different sides of the change point. A section of the GLR statistic is shown in Figure 3.13, which shows the data around the change point in Figure 3.12. As the sampling frequency is $f_s = 10^4$Hz and the length of the data windows is $d = 200$, the highest GLR value will be obtained when the former data window W_2 contains data sampled after the change, that is $t = d \cdot 1/f_s = 0.02$s after the change happens. In Figure 3.13, we can see that the maximum value of GLR is obtained at about 1.02s and as the change happens at $t = 1$s, the result obtained from simulation coincides with the expectation.

3.5.2 Two sample t-test

An alternative hypothesis test that can be used for tap change detection is the two sample t-test, which is a statistical hypothesis test that follows a Student's

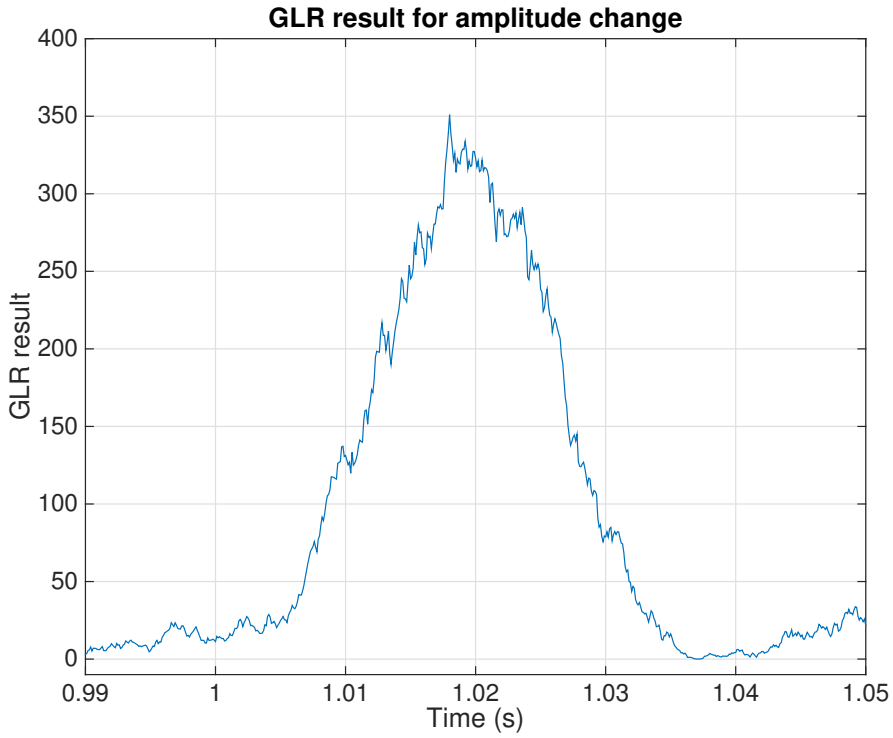


Figure 3.13: GLR result for change detection, which shows the detail around the change time point in Figure 3.12

t-distribution if the null hypothesis is supported [131, 132]. It can be used to determine if two sets of data are significantly different from each other, and is applied when the distribution of the data follows a Gaussian distribution. A two-sample t-test for a difference in mean involves independent samples or unpaired samples, which allows for the case when the sample sizes for the two samples may not be equal. The formula for unpaired two sample t-test is [131]

$$T = \frac{\hat{Y}_1 - \hat{Y}_2}{\sqrt{\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2}}}, \tag{3.57}$$

where N_1 and N_2 are the sample sizes, $\hat{Y}_1$ and $\hat{Y}_2$ are the sample means, and s_1^2 and s_2^2 are the sample variances.

If a given difference in the mean between the two example is assumed, then

the following equation can be used

$$T = \frac{\hat{Y}_1 - \hat{Y}_2 - d}{\sqrt{\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2}}}, \tag{3.58}$$

where d is the assumed difference of mean between the two sample, and T follows a t-distribution with $v = N_1 + N_2 - 2$ degrees of freedom, which provides a confidence interval that we can judge the difference between the means of two sets of data is not d. In other word, the confidence interval result obtained from two sample t-test gives the probability that no change happens.

The classical two sample t-test, however, is not an on-line method, so we also need to adapt this test to make it suitable for on-line approach. Similar to the implementation of the on-line GLR test, the parameters needed to be saved along with the variance of the data and we also need to group the data in the moving data window. Using the same simulating input data as we did for the GLR test to test the properties of two-sample t-test the data window of two sample t-test was chosen to be $d = 200$ as well. The result is shown in Figure 3.14 and Figure 3.15 (Figure 3.15 shows the detailed result around the change point). According to the result of the two sample t-test, we can judge that a change occurs at about $t = 1.02$s. The on-line two sample t-test will also detect the change after a time delay due to the length of the window, where the delay time is $t = d/f_s = 0.02$s. With the information obtained from Figure 3.14 and Figure 3.15, we can infer that there should be a change happens at $t = 1$s which coincide with the change time of the input simulating data we generated.

Comparing the simulation result of two sample t-test to that of GLR test, we find that when no change occurs, the result of two sample t-test is quite clear, while the result of GLR test is fluctuating. Also, two sample t-test gives us a straightforward result of confidence interval for change, which can be understood easily. In the GLR test, however, the result is a likelihood ratio which cannot show change probability directly and need to be compared with a threshold calculated from the model of the change.

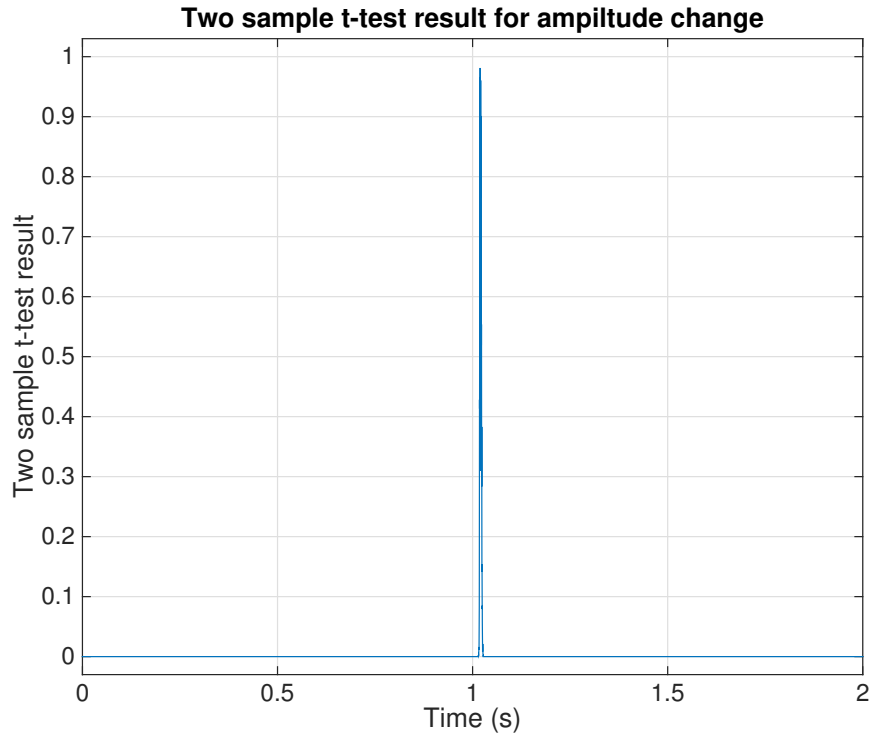


Figure 3.14: Two sample t-test result for change detection. There is a change in the input at $t = 1$s, and the two sample t-test statistic appears a peak when the change occurs.

3.6 Simulation and discussion

In order to test the performance of PLL, QRD-WRLS and hypothesis test, we need to apply them to both of simulated data and real-time voltage data input. The real-time voltage data was obtained by Scottish and Southern Energy (SSE) plc., Reading, and recorded the mains voltage signal of one phase over the period from 12th March 2014 to 14th March 2014. The data was obtained from an energy logger designed by the power group of the University of Oxford, with the sampling frequency of $f_s = 10$kHz. In addition, a list of the times that tap changes were made during this period was also provided.

3.6.1 PLL result with real input

Figure 3.16 shows the estimated frequency from the classical PLL and the HTPLL for the real input signal. In this figure, the blue line is the output of classical PLL,

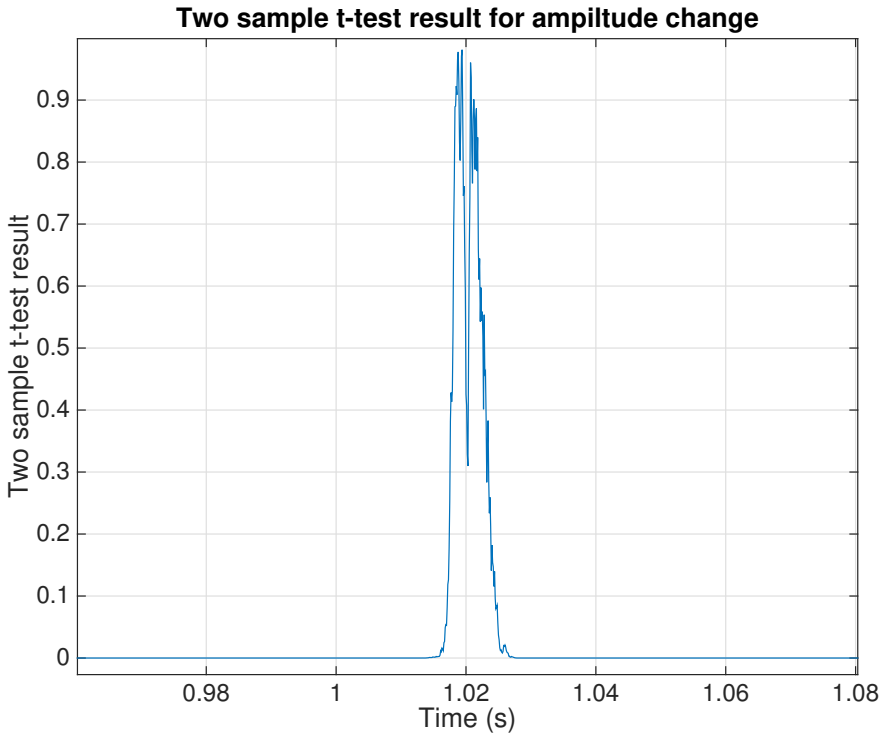


Figure 3.15: Section of two sample t-test result for change detection in Figure 3.14, showing region around change point

whereas the red line is the output of HTPLL output.

From the figure we can see that when used to track the frequency of real input signal, the output of classical PLL and HTPLL are both fluctuating. However, the fluctuation range of the classical PLL output is about 8.5Hz, which is far bigger than that of the HTPLL (which is only about 0.5Hz). We can also see from the figure that the fluctuation of classical PLL output is actually a distorted sine wave, while the fluctuation wave of HTPLL output is an irregular, but periodic wave. The period of classical PLL and HTPLL output signal's fluctuation wave is both about 0.01s, corresponding to a frequency around 100Hz, which is twice of the frequency of the input signal. For a classical PLL, this is caused by the second order harmonic in PD output, while in the HTPLL output, this is caused by the high harmonic content in the input signal. When processing the Hilbert Transformer, the high harmonic contents are also delayed for $\pi/2$, which will make the error in its output to be also a periodic function and due to the symmetrical property

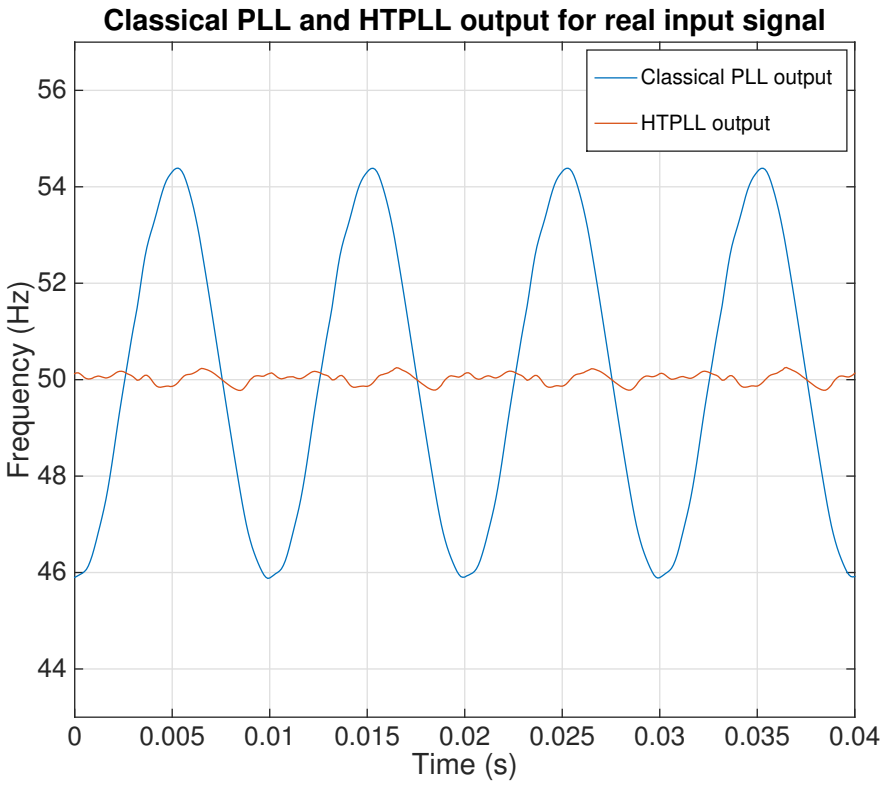


Figure 3.16: Traditional PLL and HTPLL output frequency for the same real input data. The classical PLL output is fluctuating strongly due to the second order harmonic contents whereas the HTPLL output is clearer.

of the high harmonic contents, frequency of this periodic wave should be twice of the fundamental frequency. Overall, the HTPLL's output frequency fluctuates less and its frequency tracking ability is good, confirming that the HTPLL is the better frequency tracking method for our research.

3.6.2 QRD-WRLS algorithm result with real input

For real input WRLS algorithm implementation, we choose to use the QRD-WRLS algorithm, since it can avoid the numerical problems of the classical WRLS method. Combining this with the HTPLL, we can estimate the amplitude of the input signal. The block diagram of the combined HTPLL and QRD-WRLS algorithm is shown in Figure 3.17

As shown in the diagram, the input voltage signal V_a is first put into HTPLL, then the output of PLL ϕ_o and the corresponding voltage value V_a is used in

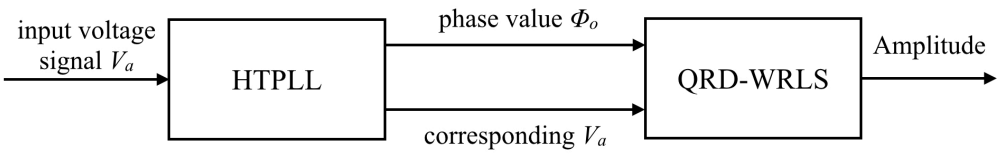


Figure 3.17: Diagram of the combined algorithm of HTPLL and QRD-WRLS

the QRD-WRLS algorithm, which generates an estimate of the amplitude of the input sine wave.

Figure 3.18 shows how the HTPLL and QRD-WRLS algorithm work together. As they both calculate data in a sliding window, the updating of the data in the two parts of the algorithm needs to be coordinated. When a new data point from the input data stream arrives, the HTPLL sliding window will move first, as shown in Figure 3.18(a), and take the new data into calculation, which provides an estimate of ϕ_o for the data point locating at the middle of HTPLL window, as illustrated in Figure 3.18(b). The next step is to update the QRD-WRLS window and move it forward to accept the data with the new ϕ_o. As shown in Figure 3.18(c), the QRD-WRLS algorithm will then run on the updated window and find the updated amplitude value. Since these algorithms are running on data windows, they are actually calculating the average result, which will lead to a delay on the final result that is half of the sum of all the windows' length. In our problem, we choose the data window to be $W_H = W_q = 600$, so the overall delay for the combination of HTPLL and QRD-WRLS algorithm should be $W_H/2 + W_q/2 = 600$ points. For our sampling frequency of $f_s = 10^4$Hz, the result will have a time delay of $600/10^4 = 0.06$s.

Figure 3.19 shows a section of data in Figure 3.11 where a tap change is known to occur. The tap change happens at around $t = 125.6$s and causes the change in the amplitude of the voltage of 3V. The blue line is the top part of the input and the red line is the output of the QRD-WRLS algorithm, which is the estimate of the input signal's amplitude.

After a tap change occurs, the input signal amplitude increases, and the output of the algorithm shows a similar change. From Figure 3.19, we can see clearly that

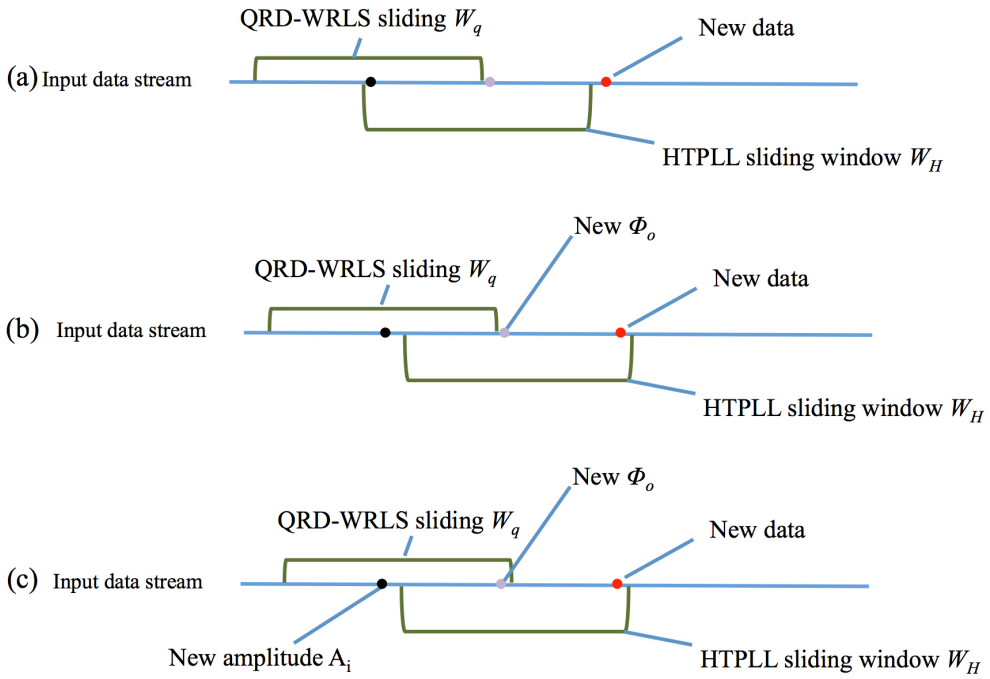


Figure 3.18: HTPLL and QRD-WRLS sliding window

there is a delay of approximately 0.05s in the output comparing to the input signal. This time delay coincides with the delay within the algorithm, and the delay time is close to the expected 0.06s. The output amplitude shows very similar change as the input signal amplitude, which means this algorithm provides information about the change of the input signal's amplitude.

It can also be noticed that there is a gap between the maximum value of the input sine wave signal and the estimated amplitude and the calculated amplitude value is bigger than the actual input signal amplitude. This is due to the high frequency harmonic contents in the signal. If we apply a Fast Fourier Transform to the signal to examine the harmonic contents, then the result is shown in Figure 3.20.

From the FFT plot, we can see that the amplitude of the signal mainly consists of the basic frequency signal which is $f_1 = 50$Hz. However, we can also see that there is high harmonic content in the signal. From Figure 3.19, which zooms in on these higher frequencies, we can see that the gap between the input signal amplitude and the calculated amplitude estimation is about 3V, which is mainly caused by the

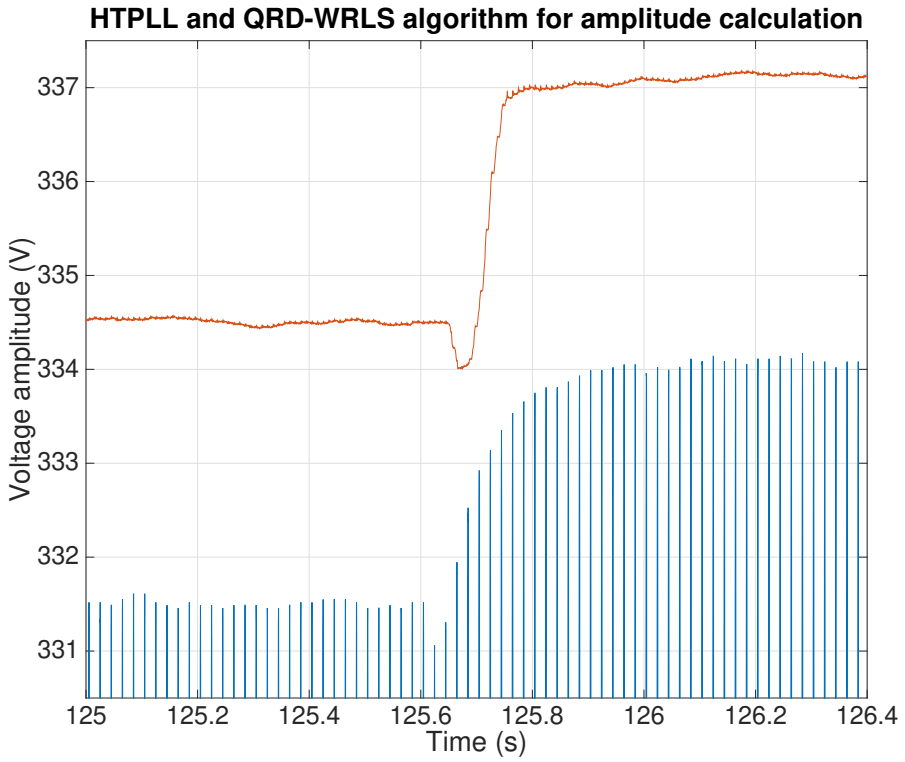


Figure 3.19: HTPLL and QRD-WRLS for amplitude calculation. The blue curve shows the top portion of the sinusoidal input voltage, and the red curve shows the combined algorithm output. The combined algorithm provides an estimation of the magnitude of the input signal.

high frequency harmonics. If we include the high frequency contents in QRD-WRLS algorithm into amplitude estimation, we can definitely achieve a better estimate of the amplitude. The red line in Figure 3.22 shows the calculated amplitude estimation including up to the 25th order harmonic and it can be seen that with high harmonic amplitude included in the calculation, an improved estimate of the input signal amplitude is obtained (note the scale on the vertical axis). The high harmonic signal in the power transmission system are mainly influenced by the demand side, which means they are not useful for detecting tap change, which is controlled by the tap change transformer from the supply side. However, since we are mainly interested in detecting the change rather than the amplitude, for our problem, applying the QRD-WRLS algorithm to estimate the first harmonic amplitude is sufficient.

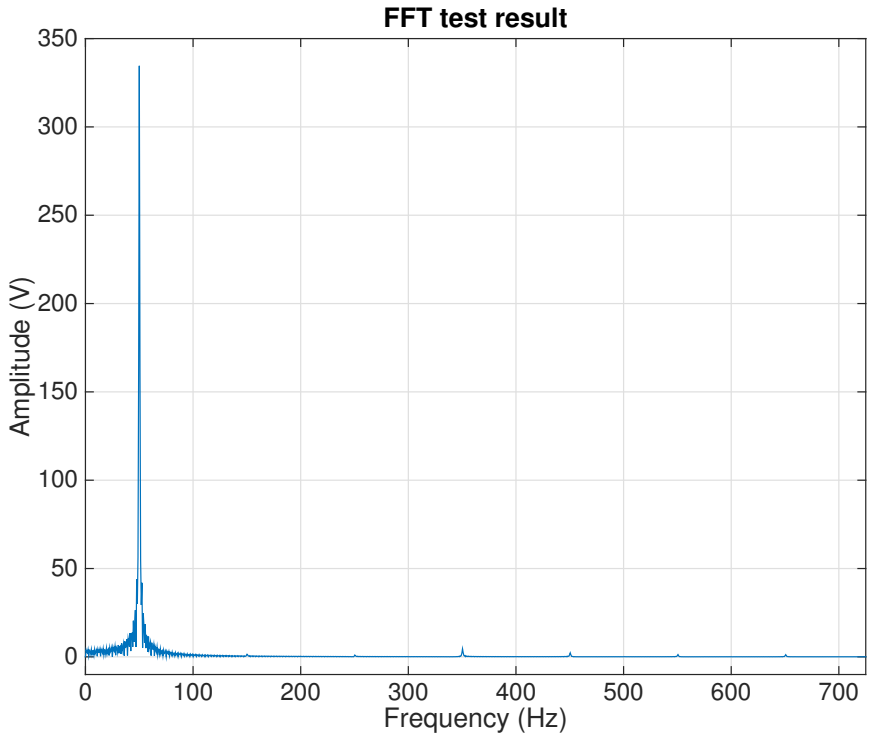


Figure 3.20: FFT result of the input voltage data in 1 second.

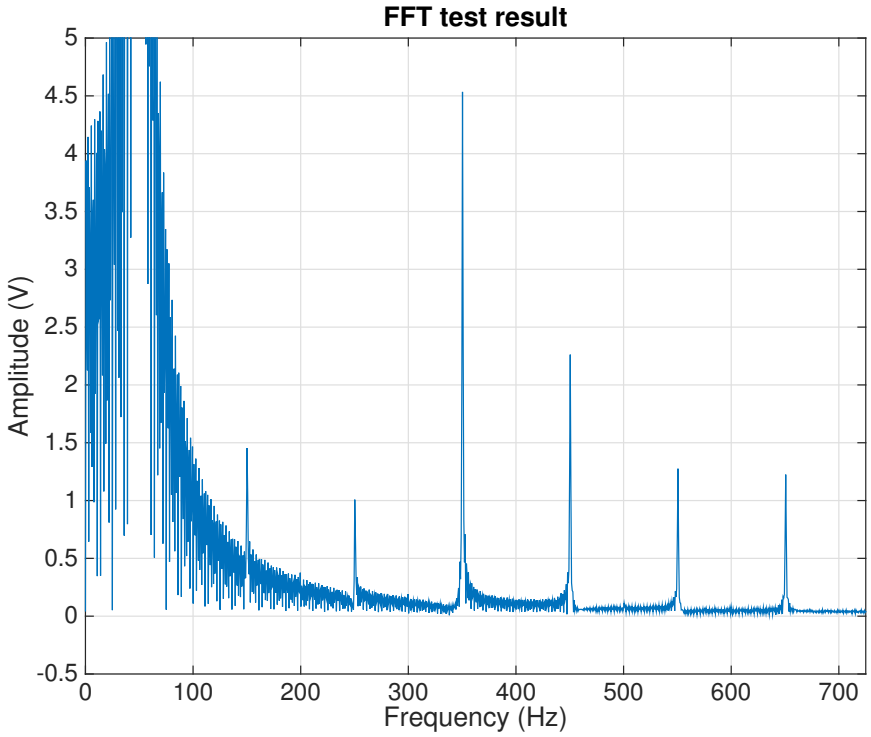


Figure 3.21: Detail of the amplitude of high harmonic contents in Figure 3.20

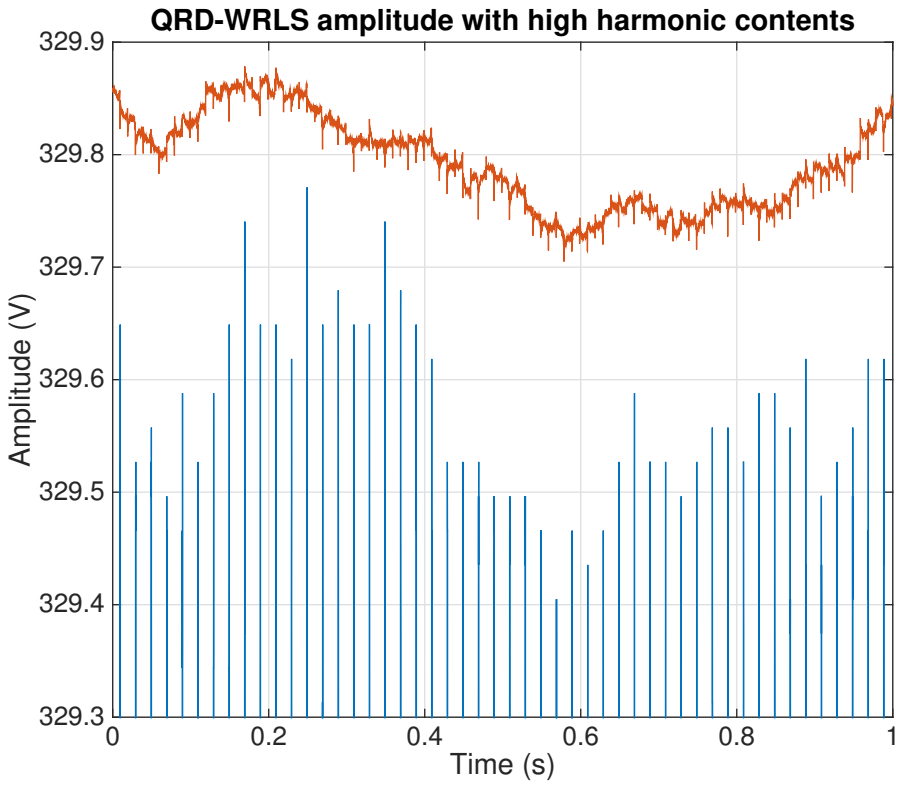


Figure 3.22: QRD-WRLS output including up to 25th harmonics, which provides a good estimate on the input voltage magnitude.

3.6.3 Hypothesis test result

The hypothesis test is applied to the QRD-WRLS output in order to identify whether a tap change has happened and we use the amplitude of the sine wave signal A as the input to the hypothesis test. For on-line implementation, the hypothesis test also need to be run within a sliding window, so there will be three sliding windows in our algorithm: PLL sliding window, WRLS sliding window and hypothesis test sliding window. These windows need to move in the correct order so that the result can be updated at each time step.

As shown in Figure 3.23, the sliding windows of HTPLL and QRD-WRLS are moving as illustrated in Figure 3.17. The difference between Figure 3.17 and Figure 3.23 is that the movement of WRD-WRLS window is included. Step (a) to (c) are exactly the same with that of Figure 3.17. Step (d) occurs when the HTPLL sliding window W_H and QRD-WRLS sliding window W_q have moved forward and

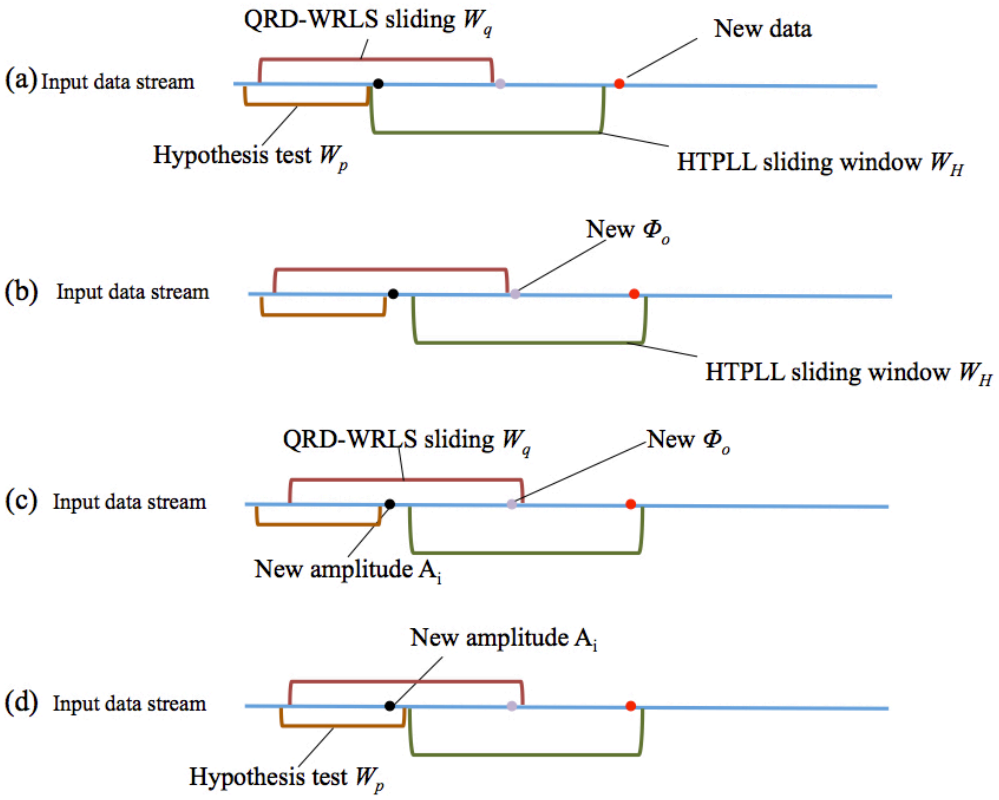


Figure 3.23: Sliding window movement order

the new amplitude A_i is calculated. Then the hypothesis test sliding window W_p is moved forward to calculate the hypothesis test statistic in order to determine the probability that a change has happened in W_p. Additionally, as these algorithms are based on sliding windows, the delay of the algorithm will be half of the sum of the length of all the sliding windows, which should be $S_d = (l_{W_p} + l_{W_q} + l_{W_H})/2$ sample, so the combined time delay will be $T_d = S_d/f_s$.

3.6.4 GLR test result

The GLR test is used to calculate the generalized likelihood ratio of the data in sliding window. We need to calculate the GLR result and the threshold in order to tell whether there is a change.

When using GLR test as the hypothesis test, we have the following hypothesis

H_0 : no change happens in the sliding window W_p

H_1 : a jump of magnitude ν happens in the sliding window W_p

In order to judge if H_1 has occurred, we need to find the threshold for this GLR test. Because a tap change will change the amplitude by about 1%, and because it requires a finite time for the tap change to occur, the change occurs over a period of about $t = 0.05$s. We can model the magnitude of a change, thus we can calculate the minimum GLR value for a tap change. The length of the tap change corresponding to about 3.5 periods of the sinusoidal input signal and the two smaller sliding windows contain $n = 200$ sample each. When a tap change occurs, we can assume that the data in the sliding window is increasing steadily and the common difference d is

$$d = \frac{k\,A}{t\,f_s}. \tag{3.59}$$

So the GLR statistic for a tap change should be

$$g_t = \frac{(3n-2)n^2\,d^2}{6R}, \tag{3.60}$$

where the R is the variance of the distribution of the underlying data. For real tap change, the change process is more abrupt, so the result calculated from (3.60) is the minimum GLR value associated with a tap change. When the GLR value exceeds g_t, we can assume that a tap change has occurred.

Figure 3.24 is the GLR result for the real tap change data from Figure 3.11, the blue line is the amplitude of the input signal which is obtained form the HTPLL and QRD-WRLS algorithm, the yellow line is the threshold for GLR test and the red line is the GLR output value. From Figure 3.19, we can see that the tap change happens at about $t = 125.6$s in the input data. From Figure 3.24 we can see that the QRD-WRLS algorithm reacts to the change about 0.05s after the tap change happens in the input and the GLR test result exceed the threshold at about $t = 125.7$s. We can also see from the figure that when there is no tap change, the GLR result is quite small and fluctuating within a negligible range.

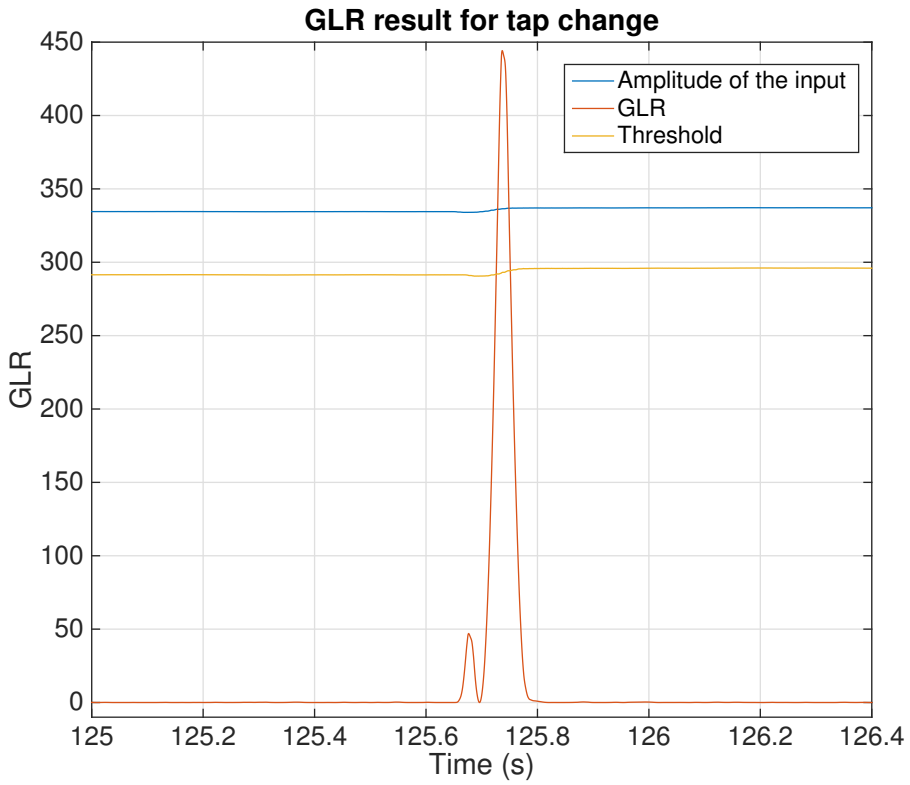


Figure 3.24: GLR result with real tap change input. When the tap change occurs at around $t = 125.6$, the GLR statistic exceeds the threshold quickly.

Table 3.1 shows the result of using the GLR test to detect the tap changes with real-time input data. The data was obtained from SSE (the data covers the single phase voltage value from 6:00 to 10:00 in the morning and 17:00 to 24:00 in the afternoon). These data was recorded on 12th March 2014, 13th March 2014 and 14th March 2014. The time of actual tap changes was also recorded.

The first column shows the recorded date of the data, the second column shows the time when the algorithm detects a tap change and the third column shows if this is a correct result. In the table, there are 19 tap changes reported by the algorithm, and 7 of them are incorrect detections. The accuracy of the algorithm is 63%. Comparing the record of the OLTC transformer, all the tap changes are successfully detected by the GLR algorithm, although there are also some incorrect detection results. In the results shown by the table, all the incorrect detections are type I errors (they are not actually tap changes but are recognised as tap

Table 3.1: Real time result for GLR test

Time	Output tap change time	Correct detection
12 March 2014	06:56:17	√
	08:09:51	×
	09:30:39	×
	17:26:06	√
	21:09:57	√
	22:48:38	√
13 March 2014	07:02:38	√
	08:23:30	×
	09:07:34	√
	09:08:30	×
	09:12:40	×
	22:11:52	√
	23:10:35	√
14 March 2014	06:38:26	√
	08:08:21	×
	08:11:07	×
	17:24:10	√
	21:13:39	√
	22:57:21	√

changes). The incorrect detection is caused by the reason that in the real input data, the voltage sometimes fluctuates strongly due to large changes in the load, which causes large changes in the GLR statistic that exceeds the threshold, so the algorithm regards these fluctuations as tap changes. The following figure shows a typical fluctuation of the voltage and most of the incorrect detection are caused by fluctuation with such shape. From Figure 3.25, we can see that the voltage first dropped for about 2.5V in about 0.06s and then went up again to the previous level quickly. Such a strong fluctuation will lead to a high GLR value in the test result. However, unlike the tap change as shown in Figure 3.6, which goes up directly, the voltage fluctuation shown in this figure goes up again after dropping, so it has a different GLR curve comparing to tap change. Comparing Figure 3.26 and Figure 3.24 we can see that the GLR curve of the fluctuation and the tap change is different. A tap change will just lead to one pulse in the GLR statistic,

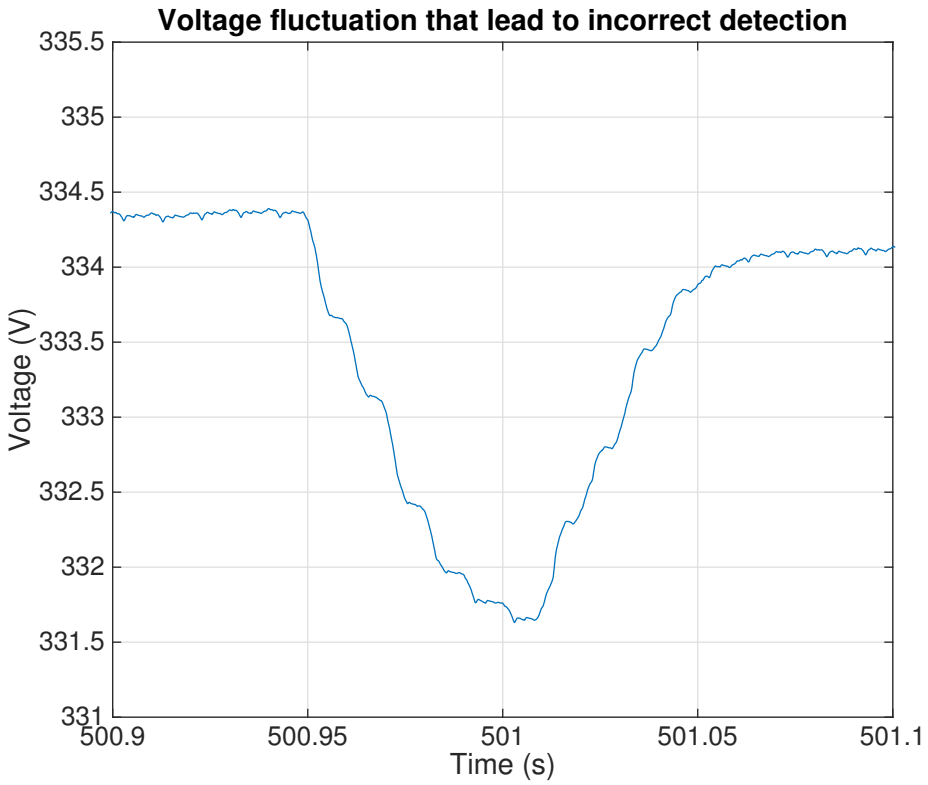


Figure 3.25: The magnitude variation of a voltage fluctuation

but a voltage fluctuation will lead to two pulses in the GLR statistic, which can be used to separate voltage fluctuation form the tap change. After adding this criteria into the algorithm, most of the incorrect detections can be removed and only one of the incorrect detection will still be recognised by the algorithm as tap change. The accuracy of the algorithm can be improved to 92%.

3.6.5 Two sample t-test result

The two sample t-test can also be used for on-line tap change detection. Similar to the implementation of GLR method, the two sample t-test should also be run with two sliding windows, since it is testing the mean difference between two groups of data. However, in order to detect the amplitude difference before and after tap change, there needs to be a gap between the sliding windows of two sample t-test, as shown in Figure 3.27, where the blue line is the amplitude of the sine wave input signal and the red line shows the sliding windows for the test. The gap between

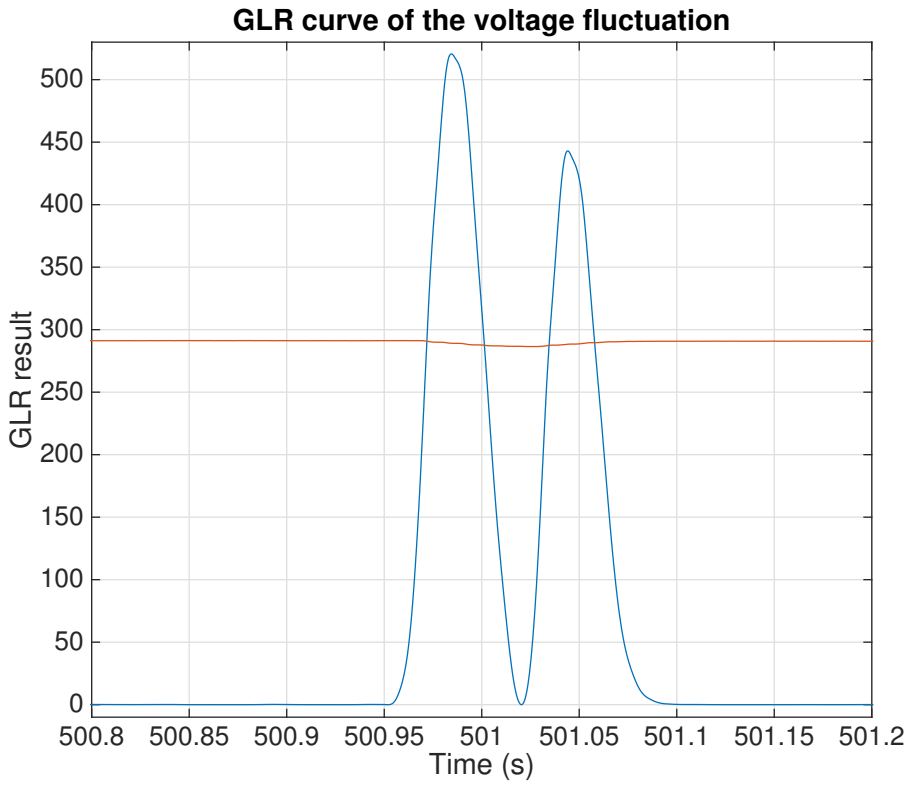


Figure 3.26: GLR curve of the voltage fluctuation, which shows a different pattern from the GLR curve of a tap change

the windows can help the algorithm to detect the change with a better accuracy. The length of the gap should approximately be the length of tap change, so that the algorithm can achieve good reaction speed as well.

Considering the time that a tap change will take, we choose the length of the gap to be 800 data, which is approximately 4 whole periods of data. The amplitude change for a tap change should be about $k = 1\%$. Then the equation for the two sample t-test (3.58) can be rearranged as

$$T = \frac{\hat{Y}_1 - \hat{Y}_2 - k\,A}{\sqrt{\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2}}}, \tag{3.61}$$

where A denotes the amplitude of the signal. The result of the two sample t-test with real tap change input is shown in Figure 3.28. From the figure we can see that there are two clear pulses indicating a tap change happens. The reason for there being two pulses in two sample t-test result for a single tap change is that

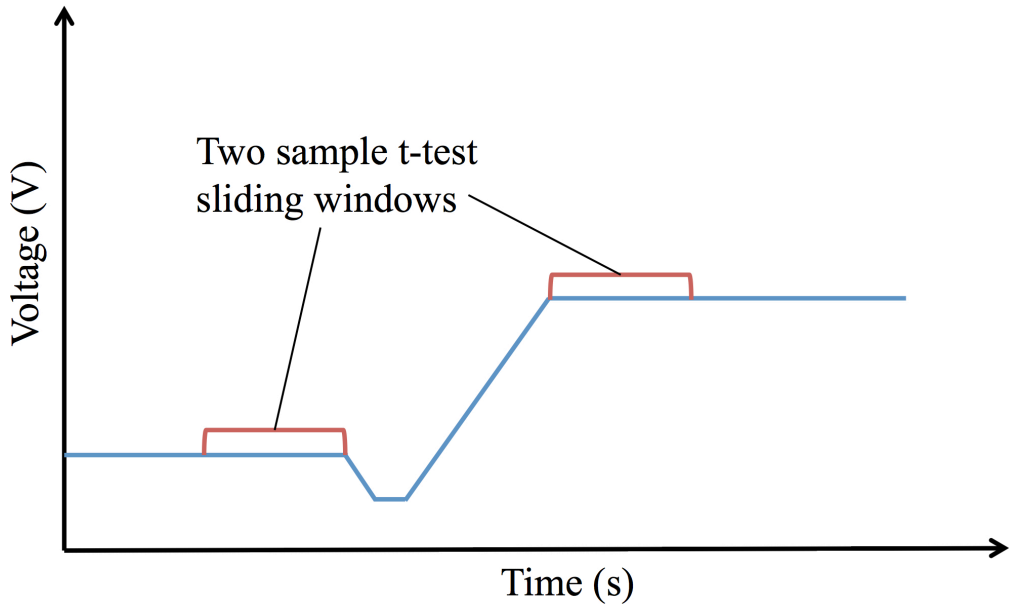


Figure 3.27: Sliding windows for two sample t-test

the expected tap change value $\Delta A = k\,A$ is smaller than the real tap change value. As it is quite difficult to predict the precise amplitude change for tap change, the expected change amount in the calculation process should be set smaller so that all the tap changes can be detected by the two sample t-test.

Table 3.2 shows the result of using the two sample t-test to detect tap change with the same real time input as used for the GLR test. Similar to the result for the GLR test, the two sample t-test can find all the tap changes from the data, but there still exist incorrect results. From the table, there are 17 tap changes reported by the algorithm, and 5 of them are incorrect. The accuracy of the detection algorithm using two sample t-test is therefore 71%. Similarly to the result of GLR test, all of the incorrect detections are type I errors (they are not tap changes but are recognised as tap changes), and they are also caused by the abrupt voltage fluctuations. Comparing the result in Table 3.1 and Table 3.2, it can be seen that the performance of the two sample t-test is much better than that of the GLR test, since it makes fewer mistakes. As with the GLR test, the two sample t-test will make incorrect detections when the voltage fluctuation is sufficiently large. Figure 3.29 shows an incorrect detection made by two sample t-test, which is caused by the voltage fluctuation shown in Figure 3.25. Comparing

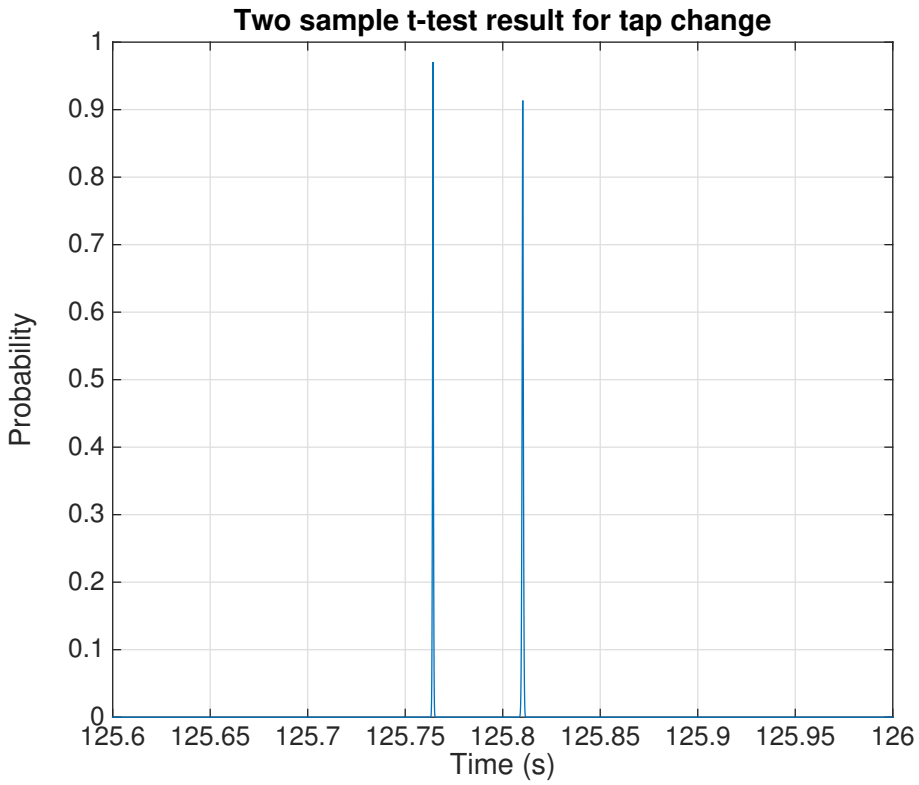


Figure 3.28: Two sample t-test result for tap change

Figure 3.29 to Figure 3.28 we can see that there will be four pulses in the two sample t-test result for voltage fluctuation, whereas for tap change, there will be only two pulses. In addition, the time gap between the pulses are also different. These differences can be used to distinguish tap change from voltage fluctuation. When this criteria is added into the tap change detection algorithm, all the incorrect detections can be removed, and the accuracy of the tap change detection algorithm can be improved to 100%. It worth to be mentioned that, if there is voltage variations vary similar to tap changes, two sample t-test may still make mistake in distinguishing these fluctuations from tap changes.

3.7 Conclusion

This chapter introduces an algorithm for the automatic detection of tap changes. The algorithm provides an identification method that can be used to indicate local network stress directly from the user side without communication to a control centre.

Table 3.2: Real time result for two sample t-test

Time	Output tap change time	Correct detection
12 March 2014	06:56:17	√
	08:09:51	×
	09:30:39	×
	17:26:06	√
	21:09:57	√
	22:48:38	√
13 March 2014	07:02:38	√
	09:07:34	√
	09:08:30	×
	09:12:40	×
	22:11:52	√
	23:10:35	√
14 March 2014	06:38:26	√
	08:08:21	×
	17:24:10	√
	21:13:39	√
	22:57:21	√

There are three main sections of the algorithm: a HTPLL is used for finding the frequency of the sinusoidal voltage signal, the QRD-WRLS algorithm is used for finding the amplitude of the signal and a hypothesis test is used for testing tap changes from the variation of the voltage amplitude.

From the results with simulated and real-time input data, it is shown that the algorithm can track the frequency of the sinusoidal voltage signal and find the amplitude of the signal and tap changes are detected by the GLR or two-sample t test. It can also be seen that this algorithm can provide good performance for tap change detection. Although it is susceptible to large fluctuations in voltage that are not associated with tap change [102], the performance of the algorithm can be improved by using better criteria.

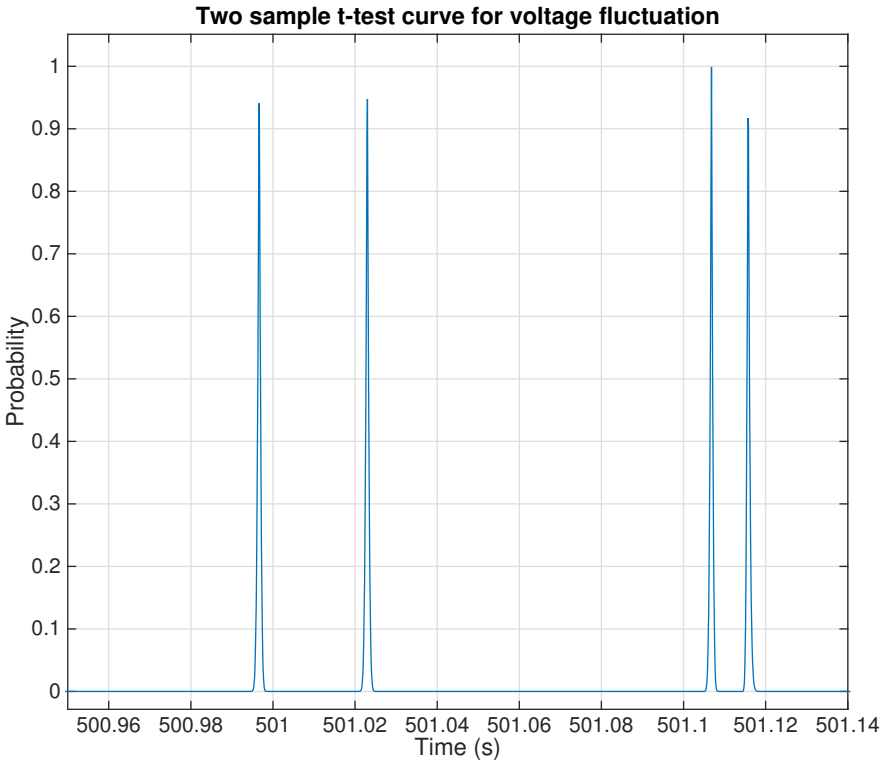


Figure 3.29: Two sample t-test curve for voltage fluctuation

4

Modelling of tap changer and power system

Contents

4.1 Introduction **121**
4.2 Power system models **122**
4.2.1 Finite generator bus model 122
4.2.2 Infinite generator bus model 126
4.2.3 ODE model of the infinite generator bus model 127
4.3 Quantized tap changer model **129**
4.3.1 Discrete tap changer model 129
4.3.2 Continuous tap changer model 130
4.3.3 Quantised tap changer model 130
4.4 Comparison of different types of models **131**
4.4.1 Finite generator bus model combined with continuous & quantised tap changer models 132
4.4.2 Infinite generator bus model 134
4.5 Removing OLTC chattering using a deadzone **136**
4.5.1 Finite generator bus model 137
4.5.2 Infinite generator bus model 140
4.6 Simulation of a 6-bus system **141**
4.7 Conclusion **146**

4.1 Introduction

Tap changes are discrete events with step changes in the transformer ratios. The effect of a tap change can be seen in Figure 4.1, which shows the top portion of the voltage signal and it can be seen that the voltage detected by a downstream load unit rapidly changes from about 331.5V to around 334V within 0.2s [68].

A number of models, spanning many levels of complexity, have been proposed for OLTCs. In general, these models are defined to be either "continuous" or "discrete" in both time and their tap positions. Discrete models are widely regarded as being the more accurate, as a tap change is a discrete event, and are commonly used when accurate simulation results are required. For example, a discrete model of a tap changer that incorporated a deadzone was used in [133] for studying the onset of limit cycles caused by the interaction between voltage fluctuation and load. A related switched model that included a dead band was considered in [134] for mitigating these limit cycles. However, for developing optimal active and reactive power controllers, as in [59], or studying voltage collapse, as in [135], these discrete models are deemed too complex as they result in hybrid systems. Discrete models are able to capture the discrete nature of a tap change, but they result in nonlinear hybrid power system models. The control of such nonlinear hybrid systems is known to be challenging, making these models unsuitable for many applications [136]. Accordingly, simpler linear models with continuous dynamics are often used, but these models have been found to give less reliable simulations [59].

A model that is able to overcome some of the inherent difficulties associated with the aforementioned models is presented in this chapter. This proposed model evolves in continuous time, but has a quantised number of tap positions. In this way, the model is both amenable to analysis whilst being rich enough to capture the essential discrete elements of a tap change.

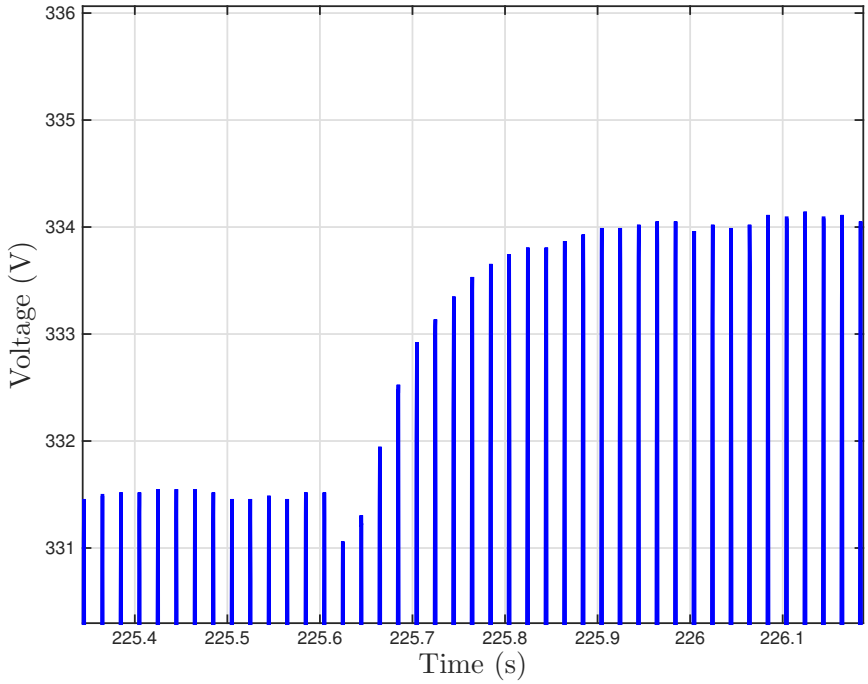


Figure 4.1: Tap change occuring within in a power network. This figure shows the the upper peak of the sinusoidal voltage signal recorded at the load side with the jump due to a tap change occurring. This information can be used to detect network stress at the user side.

4.2 Power system models

In this section, we first introduce two commonly used generator bus models. The first model considers the generator as a finite bus with the generator frequency and voltage changing according to the load. The second model assumes that the tap changer is operating at a slower time scale than the generator, with the generator being an infinite bus with the operating frequency of the system assumed to be constant. These two models are now described in greater detail.

4.2.1 Finite generator bus model

A power system model incorporating finite generator buses can be divided into three sections: generators, power transmission system and loads. Typically, the topology of this power network model is described by generators connected to load buses via a power transmission system [59]. In this model, each generator bus is assumed to

contain just one generator unit that represents the aggregation of all the generators in this regional power system, and similarly, each load bus is assumed to only have a single aggregated load. All the load buses are equipped with OLTC transformers. A key concept of this model is that the capacity of the generator is considered and thus the operating frequency of the generator and load bus will change corresponding to the variation of load. The finite generator bus model is typically used when a more realistic model of the power network is required, such as when studying the variation of the operating frequency of the generator and load bus or when studying the power flow problem. In order to have a system that is simple enough for analysis, we will consider networks containing just one generator bus and one load bus. Also, all variables describe time-varing deviations from their nominal operation point.

The finite generator bus model is derived on the basis of several assumptions.

(i) The amplitude of the generator buses are constants, that is $\forall i \in \mathscr{G}$, $v_i = 0$ where, $\mathscr{G} = \{1, ..., m\}$ represents the set of all generator buses and $\mathscr{L} = \{m+1, ..., m+n\}$ denotes the set of load buses. This is justified because the voltage magnitude of the power generators are controlled by automatic voltage regulators (AVRs) that regulate the output of the generator.

(ii) The resistance over the power transmission lines is negligible. As such, the phase δ of the voltage signal is assumed to satisfy $\sin(\delta_i - \delta_j) = 0$ and $\cos(\delta_i - \delta_j) = 1$, where $i, j \in (\mathscr{G} \bigcup \mathscr{L})$ are the units on the two sides of the power transmission cable. This assumption is based upon the almost losslessness of the power transmission lines.

(iii) The system is assumed to always be working around the operating point, which is determined by the solution of the economic dispatch problem that is solved at a slower timescale.

The finite generator bus model can be divided into two subsystems [59] with the connection of the two subsystems shown in Figure 4.2. Subsystem I represents the frequency control section, which takes the active power, load bus voltage v_i

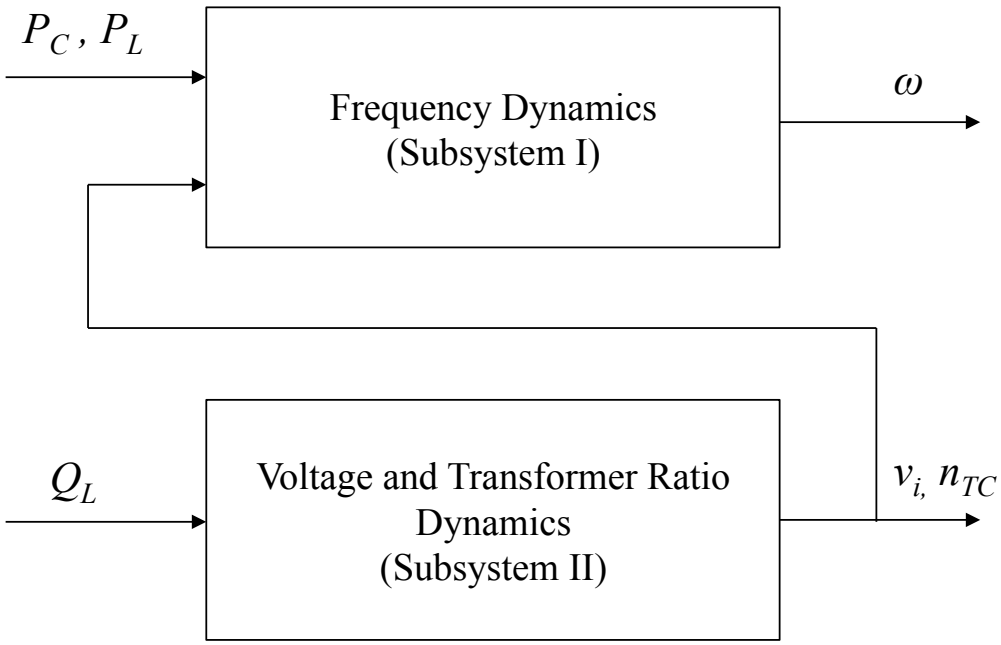


Figure 4.2: System structure of the finite generator bus model. The model can be divided into two subsystems. Subsystem I is the frequency control section and Subsystem II describes the tap changer and voltage dynamics [59].

and OLTC transformer ratio n_{TC} as the input and regulates the system frequency. Subsystem II describes the tap changer and voltage dynamics, which takes the reactive power as input and regulates the load bus voltage and OLTC transformer ratio. The dynamics of the two subsystems can be described by the following equations [5, 59, 137].

(i) Subsystem I

$$M_i\dot{\omega}_i + D_i\omega_i = P_{M_i} - P_i^{\mathscr{G}} - \sum_{(i,j)\in\mathscr{E}} T_{ij}\delta_{ij}, \quad i \in \mathscr{G}, \tag{4.1a}$$

$$D_i\omega_i = -F_i(n_i^0 v_i + v_i^0 n_i) - P_{L_i} - P_i^{\mathscr{L}} - \sum_{(i,j)\in\mathscr{E}} T_{ij}\delta_{ij}, \quad i \in \mathscr{L}, \tag{4.1b}$$

$$\dot{\delta}_{ij} = \omega_i - \omega_j, \quad (i,j) \in \mathscr{E}, \tag{4.1c}$$

$$\dot{P}_{M_i} = \frac{1}{T_{TG_i}}\left(P_{c_i} - P_{M_i} - \frac{\omega_i}{R_i}\right). \tag{4.1d}$$

Defining $T_{ij} = v_i^0 v_j^0 B_{ij} > 0$, $(i,j) \in \mathscr{E}$, then there is

$$P_i^{\mathscr{G}} = P_i + \sum_{(i,j)\in\mathscr{E}, j\in\mathscr{L}} v_i^0 B_{ij}(\delta_i^0 - \delta_j^0)v_j, \quad i \in \mathscr{G}, \tag{4.1e}$$

$$\begin{aligned} P_i^{\mathscr{L}} = P_i + & \sum_{(i,j)\in\mathscr{E}} v_j^0 B_{ij}(\delta_i^0 - \delta_j^0)v_i \\ & + \sum_{(i,j)\in\mathscr{E}, j\in\mathscr{L}} v_i^0 B_{ij}(\delta_i^0 - \delta_j^0)v_j, i \in \mathscr{L}. \end{aligned} \tag{4.1f}$$

In the equations above, $M_i > 0$ is the generator inertia, $D_i > 0$ is the damping coefficient, P_{M_i} is the mechanical power input, P_i is the disturbance factor and ω_i is the frequency deviation, $T_{TG_i}, R_i > 0$ are generator constant parameters and P_{c_i} is the control information exert on the power generator. As this model assumes that the system is operating around its nominal point, n_i^0 and v_i^0 are the initial OLTC transformer ratio and load bus voltage respectively. v_i and n_i are the load bus voltage and the OLTC transformer ratio.

(ii) Subsystem II

$$\dot{v}_i = -(\epsilon v_i^0)^{-1}[E(n_{TC_i}^0 v_i + v_i^0 n_{TC_i}) + Q_{L_i} + Q_i + L_i v_i], \tag{4.2a}$$

$$\tilde{n}_{TC} = \mathcal{TC}(n_{TC}, v_i, Q_{L_i}, Q_i), \quad i \in \mathscr{L}, \tag{4.2b}$$

where ϵ is a small positive parameter, Q_{L_i} is the reactive power, Q_i is the reactive power disturbance, L_i is a parameter relating to the structure of the system and T_{T_i} is the tap changer time constant. Equation (4.2b) describes the dynamic of OLTC transformer ratio and $\tilde{n}_{TC}$ represents a tap change, which could either be continuous or discrete in time, and $\mathcal{TC}(n_{TC}, v_i, Q_{L_i}, Q_i)$ is a function that triggers the tap change. This function depends upon n_{TC}, v_i, Q_{L_i}, Q_i. Each of the models corresponds to different $\tilde{n}_{TC}$ and $\mathcal{TC}$.

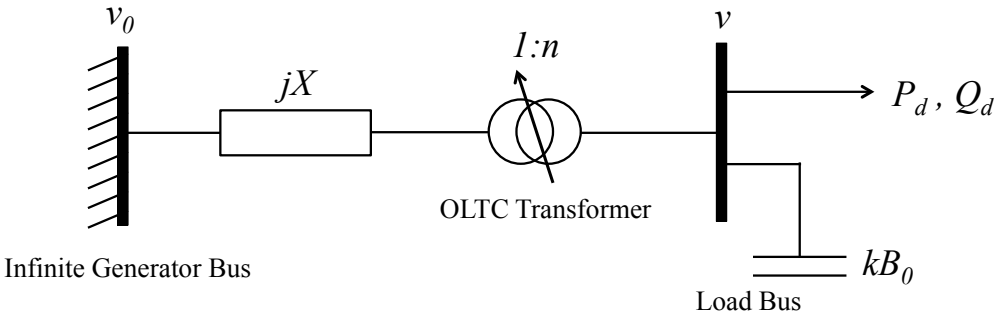


Figure 4.3: Diagram of a simplified power system network with a single infinite generator bus. In this simplified system, there is only one generator bus and one load bus, which are connected to each other via a power transmission system equipped with OLTC transformer.

4.2.2 Infinite generator bus model

The second class of generator bus model that is typically used in tap changer modelling treats the generator as an infinite bus. Such a model for a single generator bus, single load bus power system is shown in Figure 4.3, where an OLTC transformer lies between the power transmission cable and the load bus. In this power system model, the generator bus supplies electricity power for a local power distribution system through the OLTC transformer. Such a model is used when studying the interaction between the OLTC transformer and the loads.

The load bus is modelled as an exponential recovery load with the following dynamic equations [133, 138]

$$\dot{x}_p = \frac{1}{T_p}\left(-x_p + P_s(V) - P_t(V)\right), \tag{4.3a}$$

$$P_d = x_p + P_t(V), \tag{4.3b}$$

where x_p is the active load status, T_p is the active power recovery time constant, $P_s(V)$ and $P_t(V)$ are the steady-state and transient voltage dependency respectively [138, 139]

$$P_s(V) = \mathrm{k}P_0V^{\alpha_s}, \tag{4.4a}$$

$$P_t(V) = \mathrm{k}P_0V^{\alpha_t}. \tag{4.4b}$$

The reactive load power x_q is described by a similar model

$$\dot{x}_q = \frac{1}{T_q}\left(-x_q + Q_s(V) - Q_t(V)\right), \tag{4.5a}$$

$$Q_d = x_q + Q_t(V), \tag{4.5b}$$

with corresponding steady state reactive power $Q_s(V) = \mathrm{k}Q_0V^{\beta_s}$, transient reactive power $Q_t(V) = \mathrm{k}Q_0V^{\beta_t}$, reactive load status x_q and the time constant T_q. The time constant T_q is the reactive power recovery time constant. The parameter k is a scale factor on the load parameters P_0, Q_0 and B_0.

Considering the structure of the power system model and the load dynamic equations (4.3), (4.4) and (4.5) the model can be expressed as a differential-algebraic equation (DAE) [140]

$$\dot{x}_p = \frac{1}{T_p}(-x_p + \mathrm{k}P_0(v^{\alpha_s} - v^{\alpha_t})), \tag{4.6a}$$

$$\dot{x}_q = \frac{1}{T_q}(-x_q + \mathrm{k}Q_0(v^{\beta_s} - v^{\beta_t})), \tag{4.6b}$$

$$0 = \frac{v\sin(\delta)v_0}{Xn} + x_p + \mathrm{k}P_0v^{\alpha_t}, \tag{4.6c}$$

$$0 = \frac{v(v - \cos(\delta)v_0n)}{Xn^2} + x_q\mathrm{k}Q_0v^{\beta_t} - \mathrm{k}B_0v^2. \tag{4.6d}$$

Infinite generator bus models are typically used when the dynamics of the tap changer are of interest, since the variation of the generator's frequency and capacity can be neglected as the dynamics of the tap change are much slower than those of the generator.

4.2.3 ODE model of the infinite generator bus model

The model described in (4.6) is a DAE model, which is difficult to simulate it numerically due to the need to satisfy the algebraic constraints. In order to obtained a model that can be simulated and analysed easily, the DAE form of the infinite generator bus model is transformed from (4.6) into an ODE. This

is achieved by analytically solving the nonlinear algebraic equations (4.6c) and (4.6d) for given value of α_t, α_s, β_t, and β_s.

For $\alpha_t = \beta_t = 2$ and $\alpha_s = \beta_s = 0$ as is commonly adopted [133], (4.6d) can be divided by v, giving

$$\frac{\cos(\delta)v_0 n}{Xn^2} = v\left(\frac{1}{Xn^2} + x_q \mathrm{k}Q_0 - \mathrm{k}B_0\right). \tag{4.7}$$

To condense the notation, let $a_1(x_q, n) = \left(\frac{1}{Xn^2} + x_q \mathrm{k}Q_0 - \mathrm{k}B_0\right)$ and $a_2(n) = \frac{v_0}{Xn}$. An expression for v can then be obtained

$$v(x_p, x_q, n) = \frac{a_2(n)}{a_1(x_q, n)} \cos(\delta(x_p, x_q, n)). \tag{4.8}$$

Substituting this expression into (4.6c) allows it to be expressed as

$$\begin{aligned} 0 = &\left(\mathrm{k}P_0 \frac{a_2(n)^2}{a_1(x_q, n)^2}\right) \cos^2(\delta) \\ &+ \left(\frac{a_2(n)^2}{a_1(x_q, n)}\right) \sin(\delta)\cos(\delta) + x_p. \end{aligned} \tag{4.9}$$

Defining $a_3(x_q, n) = \frac{a_2(n)^2}{2a_1(x_q,n)}$, $a_4(x_q, n) = \mathrm{k}P_0 \frac{a_2(n)^2}{2a_1(x_q,n)^2}$ and $a_5(x_p, x_q, n) = x_p + \mathrm{k}P_0 \frac{a_2(n)^2}{2a_1(x_q,n)^2}$ allows an analytical expression for δ to be obtained in terms of the remaining model states, as the closed form solution of (4.9) is

$$\delta(x_p, x_q, n) = \tan^{-1}\left(\frac{a_3(x_q, n) - (a_3(x_q, n)^2 + a_4(x_q, n)^2 - a_5(x_p, x_q, n)^2)^{\frac{1}{2}}}{a_4(x_q, n) - a_5(x_p, x_q, n)}\right). \tag{4.10}$$

This equation has more than one possible solution for δ and care has to be taken in choosing the correct solution.

Using (4.9) and (4.10), an analytic expression for δ and v can be expressed in terms of x_p, x_q and n. A closed form ODE from the dynamics (4.6) can be obtained

$$\dot{x}_p = \frac{1}{T_p}(-x_p + \mathrm{k}P_0(v^{\alpha_s} - v^{\alpha_t})), \tag{4.11a}$$

$$\dot{x}_q = \frac{1}{T_q}(-x_q + \mathrm{k}Q_0(v^{\beta_s} - v^{\beta_t})), \tag{4.11b}$$

$$v(x_p, x_q, n) = \frac{a_2(n)}{a_1(x_q, n)} \cos(\delta(x_p, x_q, n)), \tag{4.11c}$$

$$\delta(x_p, x_q, n) = \tan^{-1}\left(\frac{a_3(x_q, n) - (a_3(x_q, n)^2 + a_4(x_q, n)^2 - a_4(x_p, x_q, n)^2)^{\frac{1}{2}}}{a_4(x_q, n) - a_4(x_p, x_q, n)}\right). \tag{4.11d}$$

Reducing the DEAs to a pair of coupled ODEs improves both the speed and the robustness of the simulation as it is not necessary to solve the algebraic expressions at each time step.

4.3 Quantized tap changer model

This section introduces the tap changer models used in power system modelling and the proposed quantized tap changer model is described as well. The dynamics of an OLTC in operation can be quite complex [135], typically being nonlinear and involving time delays. As described in the introduction, tap changer models are generally either discrete in time and transformer ratio values, or continuous. These tap changer models both have their disadvantages. The continuous models tend to be linear and are relatively simple. By contrast, the discrete models are generally more complex, typically resulting in hybrid systems, although they are more representative of true tap change behaviour. In this section, we will introduce the commonly used tap changer models first and then the quantised tap changer model will be described in detail.

4.3.1 Discrete tap changer model

One of the most popular methods for tap changer modelling is expressed in discrete time with a finite number of tap positions. In this model, a tap change is considered as a instantaneous jump from one tap position to another.

The discrete tap changer can be modelled as [135]:

$$n[k+1] = n[k] - df(\Delta V), \tag{4.12}$$

where $n[k]$ is the tap ratio at the k^{th} time step and ΔV is the voltage error on the controlled side, which is defined as $\Delta V = V - V_0$, where V is the voltage at the

load bus and V_0 is the reference voltage. d is the size of a single step in tap change position. Denoting the deadband of the tap changer as DB, $f(\Delta V)$ is given by [133]:

$$f(\Delta V) = \begin{cases} 1, & \text{if } \Delta V > DB \text{ for a period of } (T_d + T_m), \\ -1, & \text{if } \Delta V < DB \text{ for a period of } (T_d + T_m), \\ 0, & \text{otherwise,} \end{cases} \tag{4.13}$$

where T_d is a time delay and T_m is the mechanical switching time. It should be mentioned that both the time delay and mechanical switching time are deadband and voltage error dependent, with $T_d = T_{d_0}(DB/|\Delta V|)$ and $T_m = T_{m_0}(DB/|\Delta V|)$ respectively, where T_{d_0} and T_{m_0} are deadband design parameters. In this paper, in order to simplify the model, the time delay and the mechanical switching time of the tap changer are both neglected, so that $T_{d_0} = T_{m_0} = 0$ [133].

4.3.2 Continuous tap changer model

While being able to capture the jumps of a tap change, the discrete tap changer model of section 4.3.1 can be challenging to analyse. At the same time, each step change in tap position often only contributes a comparatively small amount of voltage correction in practise [141]. For this reason, the dynamics of the tap changer are often approximated by a continuous tap change model. This model is linear and continuous in time and assumes an infinite number of tap change positions [59, 135]

$$\dot{n}_c = \frac{1}{T_{TC}} \left(V_0 - V \right), \tag{4.14}$$

where $\dot{n}_c$ are the dynamics of the continuous tap changer model transformer ratio, T_{TC} is the time constant of the OLTC transformer, V_0 is the reference voltage for the voltage control and V is the actual voltage at load bus. The simplicity of this model opens up the possibility of developing tap change controllers [59] and studying voltage collapse in the network [135].

4.3.3 Quantised tap changer model

In this section, we introduce a new tap changer model that combines some of the benefits of both the continuous and the discrete models. This model is capable of

describing both the discrete jumps of tap change by using a quantised tap change, whilst still being analysable (as it is posed in continuous time). Being continuous means that when the model is connected with the power networks models, it does not result in a hybrid system. In essence, the proposed model is a quantised version of the continuous model of Section 4.3.2, with the quantisation function being used to enforce a finite number of tap change positions. This function $Q(x) : \mathbb{R} \to \mathcal{Q}$ rounds input data $x \in \mathbb{R}$ to the closest quantised value in the set $\mathcal{Q}$ via

$$Q(x) = \Delta \cdot \left\lfloor \frac{x}{\Delta} + \frac{1}{2} \right\rfloor, \tag{4.15}$$

where $\lfloor x \rfloor = floor(x)$ is the floor function that returns the maximum integer smaller than x and Δ is the quantisation interval.

The proposed quantised model is described by

$$\dot{n}_q^* = \frac{1}{T_{TC}} (V_0 - V), \tag{4.16a}$$

$$n_q = Q(n_q^*) = \Delta_{TC} \cdot \left\lfloor \frac{n_q^*}{\Delta_{TC}} + \frac{1}{2} \right\rfloor, \tag{4.16b}$$

where n_q is the quantised value of a intermediate variable n_q^*, T_{TC} is the time constant of tap changer, V_0 is the controlled reference voltage, V is the actual load bus voltage and Δ_{TC} is the quantisation interval, which should be equal to the tap change size. The tap changer model of (4.16) is continuous in time, the quantiser used in this model gives the discrete variation of its transformer ratio. Combining this tap changer model with the power system models in continuous time will not lead to hybrid system.

4.4 Comparison of different types of models

In this section, combinations of the two types of power system models with the three different tap changer models are simulated. These simulations represent a comparison of results from the literature [59, 133, 134, 140] and it is shown that each tap changer model gives its own unique response. These responses affect the conclusions that can be drawn about the power network simulations,

including their stability and robustness. The parameter used for the simulations are given in the tables below.

Table 4.1: Parameters for Finite Generator Bus Model [59]

Parameter	Value	Parameter	Value	Parameter	Value
M_i	5	D_i	1	T_{TG_i}	5
F_i	2	R_i	0.05	B_{ij}	22
E_i	1.5	v_i^0	1	n_i^0	1
B_i	5	T_{T_i}	0.5	ϵ_i	0.1

Table 4.2: Parameters for Infinite Generator Bus Model [140]

Parameter	Value	Parameter	Value	Parameter	Value
X	0.5	B_0	0.5	P_0	1
Q_0	0.5	$T_p = T_q$	60	$\alpha_s = \beta_s$	0
$\alpha_t = \beta_t$	2	Tap step	1%	Tap limits	$\pm$15%
$DB/2$	0.9%	T	30	k	1

4.4.1 Finite generator bus model combined with continuous & quantised tap changer models

This is the set up considered in [59] where the continuous tap change model was used so as to make the design of an optimal controller feasible. The model dynamics are given by (4.1), (4.2), (4.14) and (4.16). For the simulations, the appropriate tap changer model is substituted into the dynamics for n_{TC} in (4.2b) which is defined by the function $\mathcal{TC}(n_{TC}, v_i, Q_{L_i}, Q_i)$, with the model parameters values given in Table 4.1

The simulation is for the single generator and single load bus model, as shown in Figure 4.4. This simplified model is used to highlight the main properties of the tap changer models. The response of the continuous model was rather benign, converging towards an equilibrium with OLTC transformer ratio $n = 1.014$, load bus voltage $V = 0.9871$ and active load power $P = 0.0129$ (The simulation results

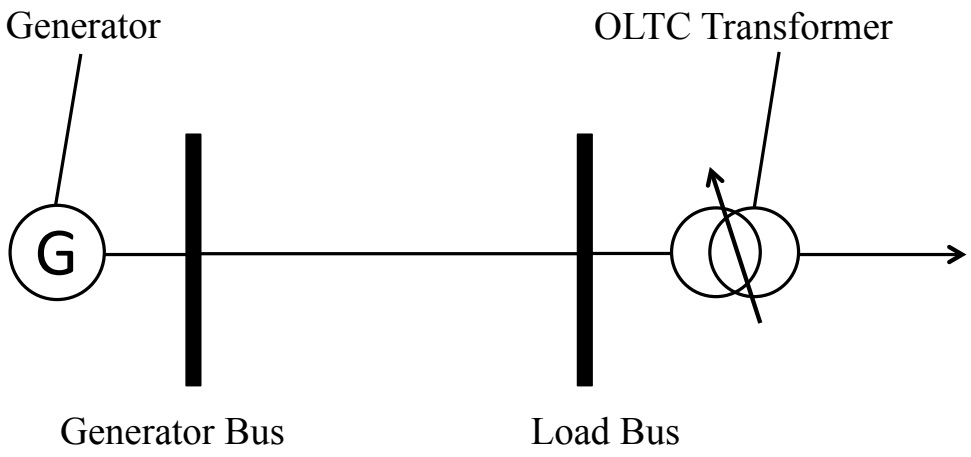


Figure 4.4: Diagram of a single finite generator bus and single load bus system of the finite generator bus model. The OLTC transformer is considered to be located within the load bus.

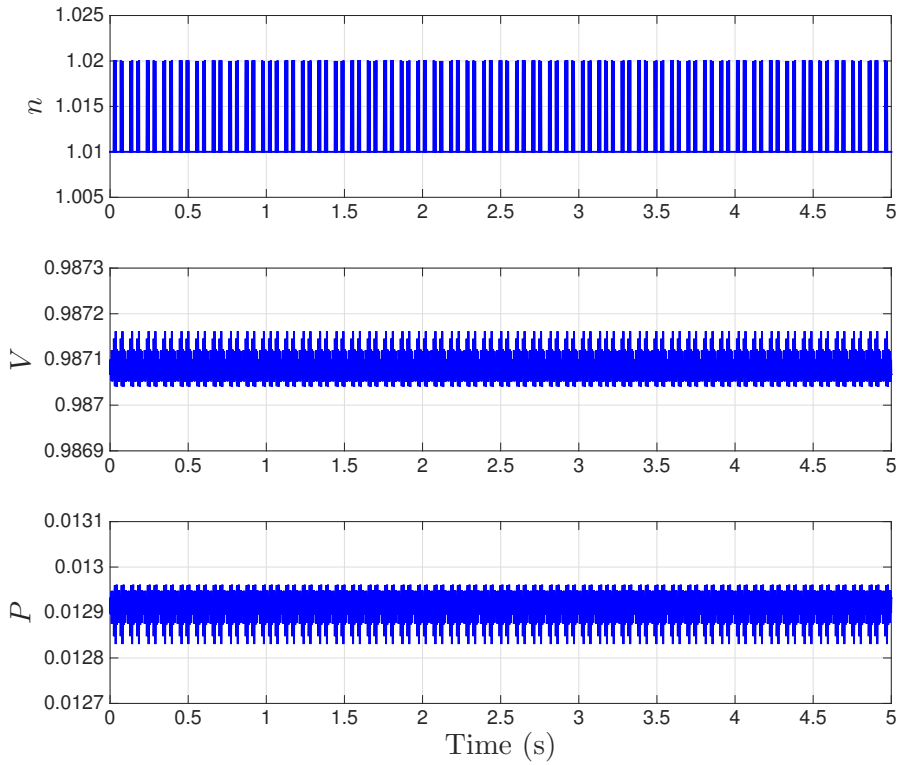


Figure 4.5: Simulation of OLTC transformer ratio n, load bus voltage V and active load power P for finite generator bus model of (4.1) and (4.2) with quantised tap changer model (4.16). The output of the system shows a pattern of chattering around its equilibrium point.

of load bus voltage V and active load power P are in *per unit* = *p.u.*). By contrast, the output of the quantised model is changing discretely, with chattering around the equilibrium point of the continuous model, as shown in Figure 4.5. The OLTC transformer ratio n and load bus voltage V in the dynamic equations (4.1), (4.2) and (4.16) represent their deviations from the nominal values $n_0 = 1$ and $v_0 = 1$, whereas in Figure 4.5, the results shown are their actual values.

Chattering in Figure 4.5 occurs because the quantised tap changer ratio cannot reach the equilibrium point, which is not well defined. A formal mathematical justification for the onset of the chattering can be obtained by comparing the model to the relay systems of [142]. The onset of chattering will damage the transformer and possibly lead to stability issues, the reason for why such chattering is not experienced in practical power systems is due to the slow response rate and time delay of the tap change. A method for removing the chattering with a deadzone will be introduced later.

4.4.2 Infinite generator bus model

The infinite generator bus model is commonly used when the load dynamics are of primary interest. It has been shown that the interconnection of this generator bus model with the discrete tap changer model can lead to dynamics such as limit cycles and other forms of bifurcations [133, 134, 138, 140].

The simulation result of the infinite generator bus model with continuous tap changer model is also benign, with the response converging towards a well-designed equilibrium point. For all times, the deviations in (4.14) equal to zero and the voltage at the load bus stays at the desired value $V_0 = 1$. The equilibrium point of OLTC transformer ratio using the parameters of Table I is $n = 0.895$. The active load status $x_p = 0$ and active power $P_d = 1.03$. The reactive load status and reactive power have similar dynamic as the active load status and active power characteristics respectively, so they have similar results.

Figure 4.6 shows the simulation with the discrete tap changer model. During the time period from $t = 8$min to around $t = 9.5$min, the tap change ratio is higher

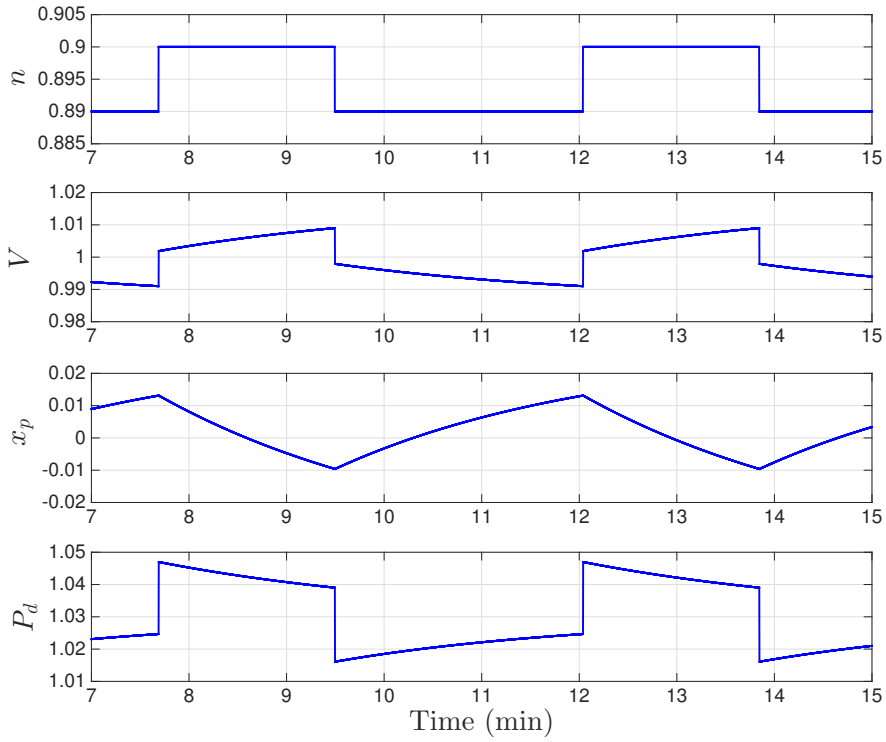


Figure 4.6: Simulation result of a single infinite generator bus model shown in Figure 4.3 connected with the discrete tap changer model of (4.13). The variation of the OLTC transformer ratio n, load bus voltage V, active load status x_p and active power P_d show a pattern of limit cycle. In the figure, V, x_p and P_d result are in per unit (p.u.).

and the voltage deviation is positive causing the voltage to increase following an exponential response. At the same time, the load status and actual active power at the load bus both decrease due to the increase of the voltage. At about $t = 9.5$min, the voltage exceeds the upper band limit and thus a step down tap change occurs. The transformer ratio changes to a smaller value and the system then moves towards the new the operating point determined by this transformer ratio. The load bus voltage starts to increase, while the load status and active power are decreasing. If the recovery performance is strong enough, as is the case shown in this figure, the voltage exceeds the lower band limit at about $t = 12$min and then a reverse tap change is made. This will result in the system cycling.

In Figure 4.7, the simulation result of quantised tap changer model with infinite generator bus is shown, where Figure 4.8 shows the detail of the chattering in

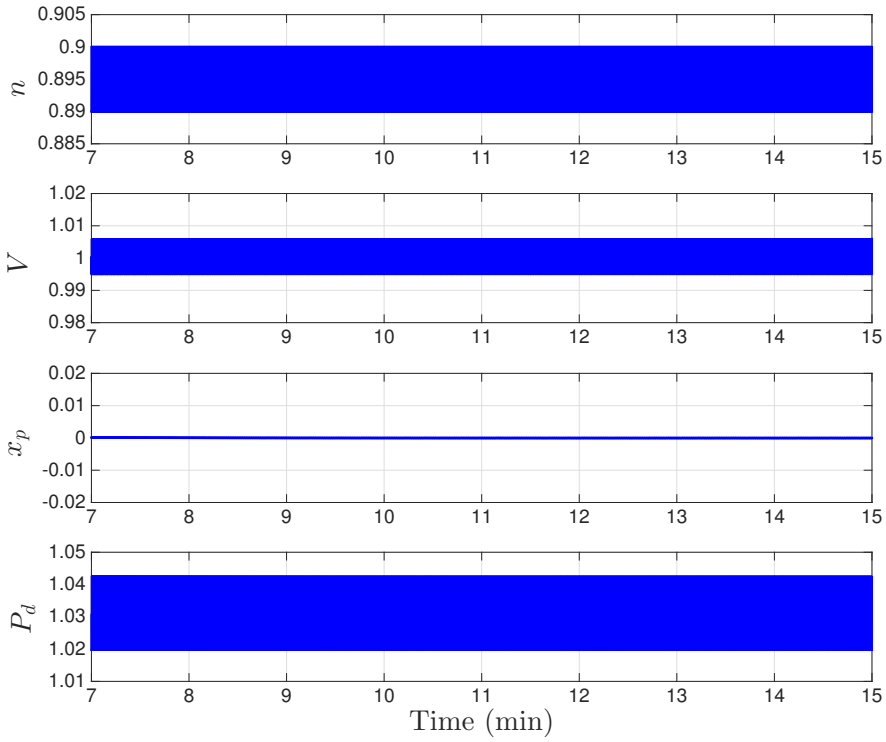


Figure 4.7: Simulation result of a single infinite generator bus model shown in Figure 4.3 connected with the discrete tap changer model of (4.16). The variation of the OLTC transformer ratio n, load bus voltage V, active load status x_p and active power P_d show a pattern of chattering around the equilibrium point. In the figure, V, x_p and P_d result are in per unit (p.u.).

Figure 4.7. Due to the introduction of the quantiser, the equilibrium point of the system cannot be reached, so the response chatters. The reason for this chattering is the same as that for the model of Section 4.4.1, and a method to remove it will now be described.

4.5 Removing OLTC chattering using a deadzone

In the previous section, it was shown that the quantised tap change model induced chattering. This response is unphysical and has consequences for the network stability and damage to the OLTC transformer. A method to remove this chattering is now proposed that is based upon the introduction of a deadzone.

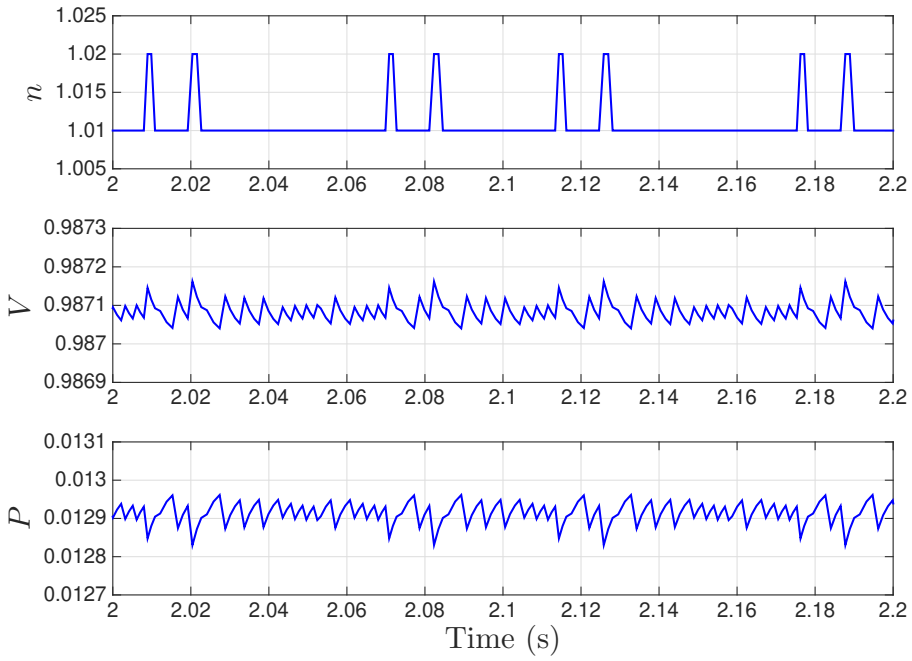


Figure 4.8: Detail of the chattering shown in Figure 4.7

4.5.1 Finite generator bus model

The chattering induced by the quantised tap changer model appears because the model equilibrium point can not be reached as it is not in the range of the quantisation values. A method to enforce the existence of a suitable equilibrium point is to process the dynamics through a deadzone with this solution shown in Figure 4.9.

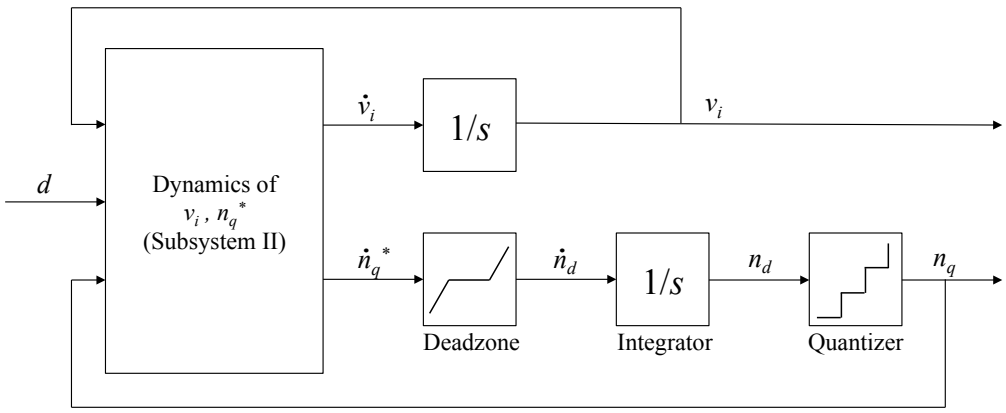


Figure 4.9: Diagram of the Subsystem II of the finite generator bus model from Figure 4.2 with the deadzone chattering removal method included. In this case, the OLTC transformer dynamics is first processed by a deadzone and then put into an integrator and quantiser.

With the deadzone included, the quantised tap changer model has the state-space form

$$\begin{bmatrix} \dot{v_i} \\ \dot{n_q^*} \end{bmatrix} = \begin{bmatrix} -(\epsilon v_i^0)^{-1} E n_{TC_i}^0 + L_i & -(\epsilon v_i^0)^{-1} E v_i^0 \\ -T_{T_i}^{-1} n_{TC_i}^0 & -T_{T_i}^{-1} v_i^0 \end{bmatrix} \begin{bmatrix} v_i \\ n_q \end{bmatrix} + \begin{bmatrix} Q_{L_i} + Q_i \\ 0 \end{bmatrix}, \tag{4.17}$$

where $n_q = Q(n_d)$, $\dot{n_d} = dzn(\dot{n_q^*})$ and $dzn(x)$ is the deadzone function

$$dzn(x) = \begin{cases} x - \lambda, & x > \lambda, \\ 0, & -\lambda < x < \lambda, \\ x + \lambda, & x < -\lambda. \end{cases} \tag{4.18}$$

In this case, when $\dot{n}_q^*$ is smaller than λ, then $\dot{n}_d = 0$ and a well defined equilibrium exist. It is therefore important to size the deadzone correctly to enforce the existence of a suitable equilibrium point that is not too distant from the desired point.

For the ease of notation, the state-space form of (4.17) is simplified to

$$\begin{bmatrix} \dot{v_i} \\ \dot{n_q}^* \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} v_i \\ n_q \end{bmatrix} + \begin{bmatrix} \bar{d} \\ 0 \end{bmatrix}, \tag{4.19}$$

where $A_{11} = -(\epsilon v_i^0)^{-1}(E n_{TC_i}^0 + L_i)$, $A_{12} = -(\epsilon v_i^0)^{-1} v_i^0$, $A_{21} = T_{T_i}^{-1} n_{TC_i}^0$ and $A_{22} = T_{T_i}^{-1} v_i^0$, with the incoming power $\bar{d} = -(\epsilon v_i^0)^{-1}(Q_{L_i} + Q_i)$ treated as a disturbance. We will consider the deadzone band limit $\lambda = D_q$ as a design parameter and obtain an analytical result for its sizing. It is noted that in the region where

$$|A_{21} v_i + A_{22} n_q| > D_q, \tag{4.20}$$

then the dynamics of the system with the deadzone are similar to those without both the quantiser and the deadzone. In the region where

$$|A_{21} v_i + A_{22} n_q| < D_q, \tag{4.21}$$

the equilibrium point of the system with deadzone and quantisation is defined. The goal now is to find the minimum value of D_q that gives a suitable equilibrium such it as close to the equilibrium point of the continuous model as possible.

To achieve this, it is noted that the equilibrium points of the continuous tap change model satisfy the linear equation

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} v_i^0 \\ n_{TC_i}^0 \end{bmatrix} + \begin{bmatrix} \bar{d} \\ 0 \end{bmatrix}, \tag{4.22}$$

while those of the quantised model with the deadzone satisfies

$$\begin{bmatrix} 0 \\ n_q^* \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} v_i \\ n_q \end{bmatrix} + \begin{bmatrix} \bar{d} \\ 0 \end{bmatrix}, \tag{4.23}$$

which is only defined if n_q equals to a quantisation interval. These two equations are linked via $\bar{d}$, and it can then be shown that (4.21) can be expressed in terms of the error in the tap change equilbrium pointns of the models, which is given by

$$\left(\frac{A_{12}A_{21}}{A_{11}} - A_{22} \right) \left(n_{TC_i}^0 - n_q \right) < D_q. \tag{4.24}$$

As the quantisation function is symmetric, $Q(x) = -Q(-x)$, then to verify (4.24), we only have to check the inequality in the range $0 \leq \dot{n}_q \leq D_q$.

Actually, defining the function $G(x)$ as

$$G(x) = Q(x) + \frac{\Delta_{TC}}{2},$$

and setting the deadzone parameter D_q as

$$D_q = \frac{\Delta_{TC}}{2} \left(\frac{A_{12}A_{21}}{A_{11}} - A_{22} \right),$$

then the output transformer ratio of the OLTC transformer quantised model with the deadzone always moves to the closest possible quantised output value.

The analysis above shows that adding a deadzone into the model can eliminate chattering, as it forces the system to have an equilibrium. However, the deadzone also causes a steady-state error in the output of the transformer ratio when compared against the continuous model.

Returning to the single load single finite generator bus system of Figure 4.4, the result of using a well designed deadzone to solve chattering problem is shown in Figure 4.10. Compared to Figure 4.7 where there is no deadzone and the output transformer ratio n_{TC_i} is constantly chattering, when a deadzone is included, the chattering problem is solved.

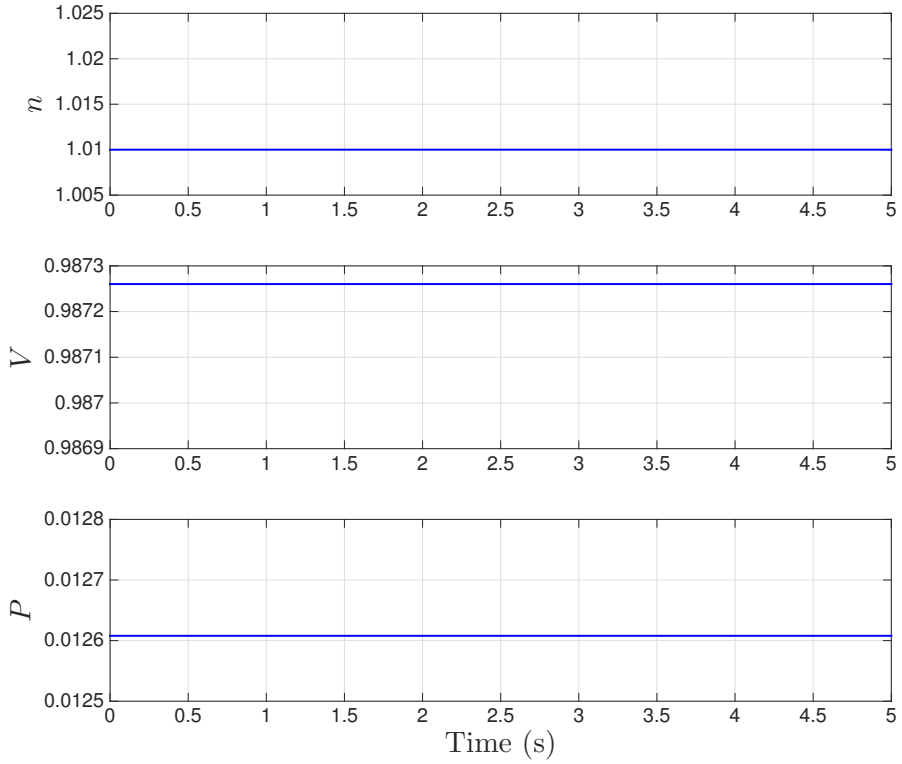


Figure 4.10: Simulation of the quantised tap change model with a well designed deadzone interconnected within the single bus finite generator network model. By introducing the deadzone, the chattering problem is solved. The transformer ratio n stays at the expected quantisation interval. The load bus voltage V and active load power P results get rid of the chattering problem as well.

4.5.2 Infinite generator bus model

As in the method introduced above, for the infinite generator bus model, a deadzone can also be added into the quantised tap changer model. With the deadzone, the dynamic equation for the tap changer model of (4.16) becomes

$$\dot{n}_q^* = \frac{1}{T_{TC}}(V_0 - V), \tag{4.25a}$$

$$\dot{n}_d = dzn(\dot{n}_q^*), \tag{4.25b}$$

$$n_q = Q(n_d) = \Delta_{TC} \cdot \left\lfloor \frac{n_d}{\Delta_{TC}} + \frac{1}{2} \right\rfloor. \tag{4.25c}$$

A simulation of this model is shown in Figure 4.11, which is free from chattering and the transformer ratio converges to its closest well designed value. Comparing this result to that of the model with a deadzone, the transformer ratio in Figure

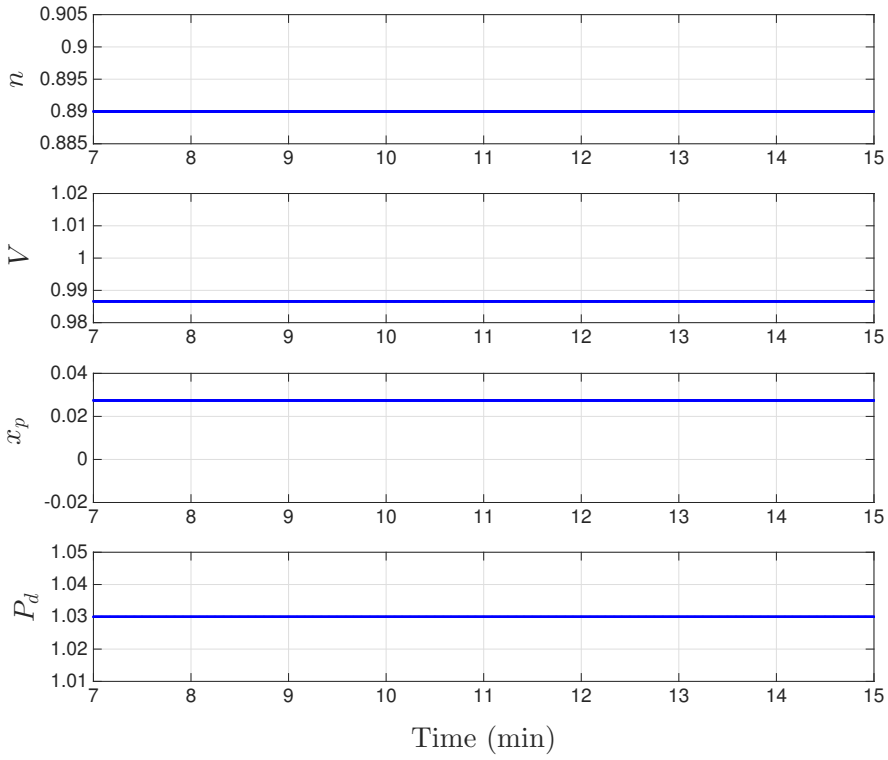


Figure 4.11: Simulation result of a single infinite generator bus model shown in Figure 4.3 connected with the discrete tap changer model with deadzone. With the deadzone, the chattering problem is solved and the transformer ratio stays at the expected quantisation interval.

4.11 stays within the quantisation interval that closest to the equilibrium of the system with continuous tap changer model.

4.6 Simulation of a 6-bus system

In this section, we provide an example to illustrate how the proposed quantised tap changer model with a deadzone can be used to study the voltage, transformer ratio and operating frequency variation in a 6-bus network such as the one shown in Figure 4.12. The frequency variations of the generator and load buses are used to indicate whether the generator and load buses are stable when there is demand change in load buses. This model has a similar topology to a power network with the load and generator buses being interconnected [59].

In this 6-bus network, *Bus* 1 and *Bus* 2 are generator buses, which are both

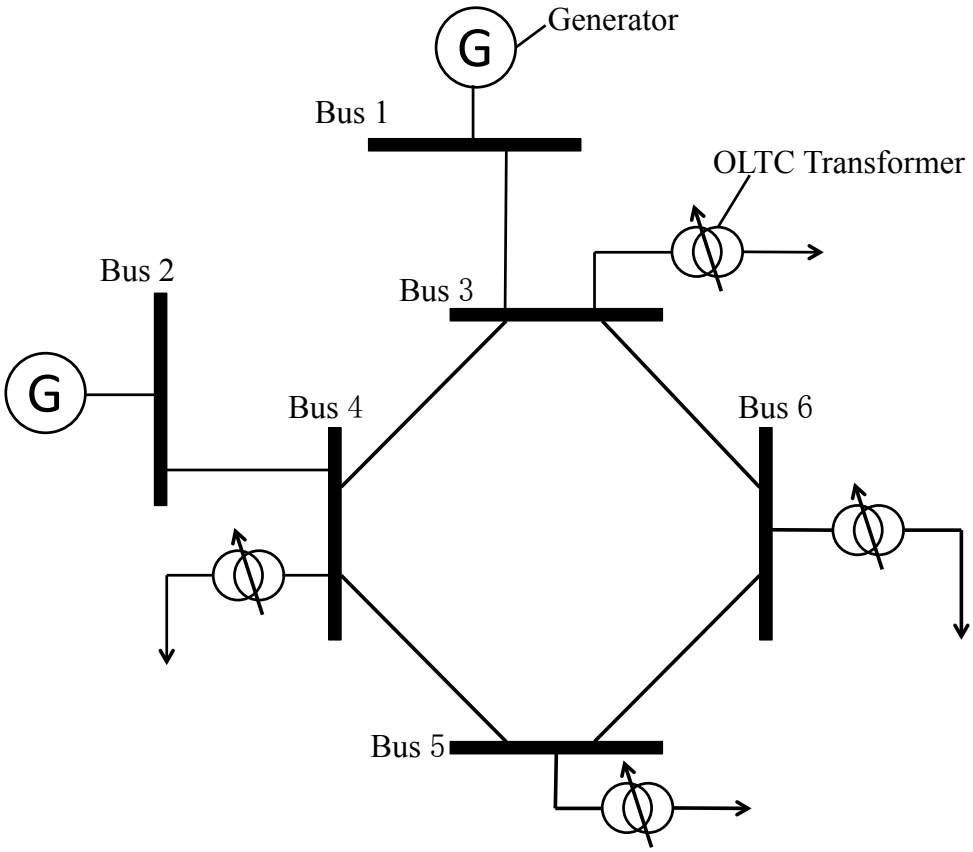


Figure 4.12: A 6-bus network [59]. *Bus* 1 and *Bus* 2 are two generator buses and *Bus* 3 to 6 are four load buses. Each load bus is equipped with OLTC transformers and are interconnected with other load buses.

modelled as finite generator buses. *Buses* 3 to 6 are load buses corresponding to the finite generator buses, each of which are equipped with OLTC transformers. The connections of the buses is also shown in Figure 4.12, with *Bus* 2 and *Bus* 4 directly connected to the two generator buses respectively. Each load bus is connected to the other load buses and has the dynamics given in (4.1). All of the OLTC transformers are modelled using the quantised tap changer model with deadzone as described in Section 4.3.3 and 4.5. The parameter values of the load buses are given in Table 4.3. The reactive power demand deviation is set as follows: for *Bus* 3, the reactive power demand deviation is 0.8 p.u. (per unit) and changes to 1.2 p.u. at $t = 80$s; for *Bus* 4, the reactive power demand is 0 p.u. and changes to 0.5 p.u. at $t = 160$s; for *Bus* 5, the reactive power demand is fixed at 0.7 p.u.; for *Bus* 6, the reactive power demand is 0 p.u. and changes to 0.6 p.u. at

Table 4.3: Parameters for 6-Bus Model [59]

Parameter	Value	Parameter	Value
$all\ n^0$	1	$all\ v^0$	1
Δ_{TC}	0.01	$all\ T_{ij}$	27

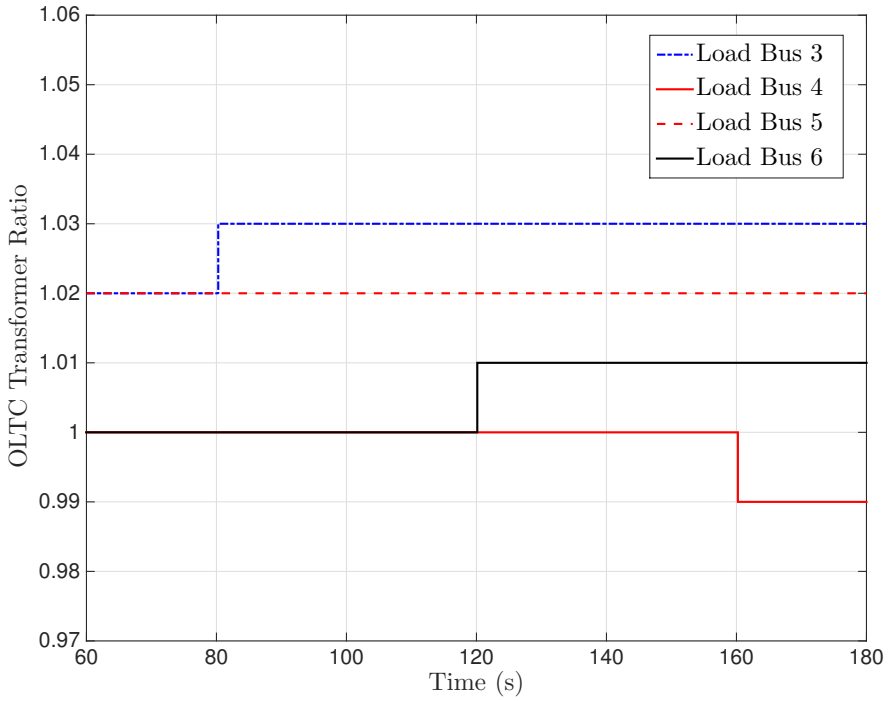


Figure 4.13: Simulation of the OLTC transformer ratio for the 6-bus network of Figure 4.12. The variation of OLTC transformer ratio in each load bus corresponds to the change in reactive power demand.

$t = 120$s. The other parameters are given in Table 4.1.

Figure 4.13 shows the simulation result of the transformer ratio. After the reactive power demand change, the OLTC transformer at the load bus makes a tap change to react to the power demand variation and the figure shows that the OLTC transformer ratio changes accordingly in a discrete manner.

Figure 4.14 shows the simulation result of the load bus voltage with the voltage of each load bus changing according to OLTC transformer ratio change. It should be mentioned that due to the difference between the parameters of each bus, each

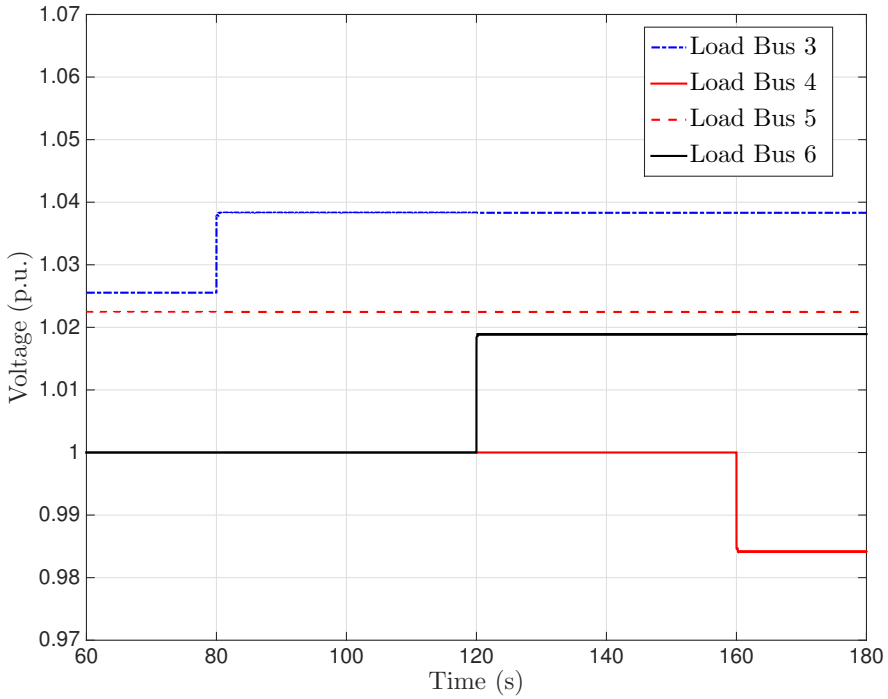


Figure 4.14: The voltage for the simulation of the 6-bus network of Figure 4.12. When the OLTC transformer ratio at each load bus changes, the voltage also changes.

bus's voltage change is different, even though there is only a single step up/down tap changer. Also, because of the connection between the load buses, the reactive power change in a certain load bus can slightly affect the voltage of other load buses.

The operating frequency deviation is shown in Figure 4.15. When a reactive power demand change causes tap changes at $t = 80, 120, 160$s, the operating frequency of the buses will fluctuate before converging to the original value of 50Hz. Looking in detail at the the frequency fluctuation, it can be seen that the largest spikes occur in the buses with the reactive power demand change. However, the fluctuations in frequency are always way much smaller than the acceptable range of 1%. Noting the scale of the axis in Figure 4.15, it can be seen that the frequency variation of the generator buses are even smaller. This suggests that the 6-bus network is stable when subject to a reactive power change.

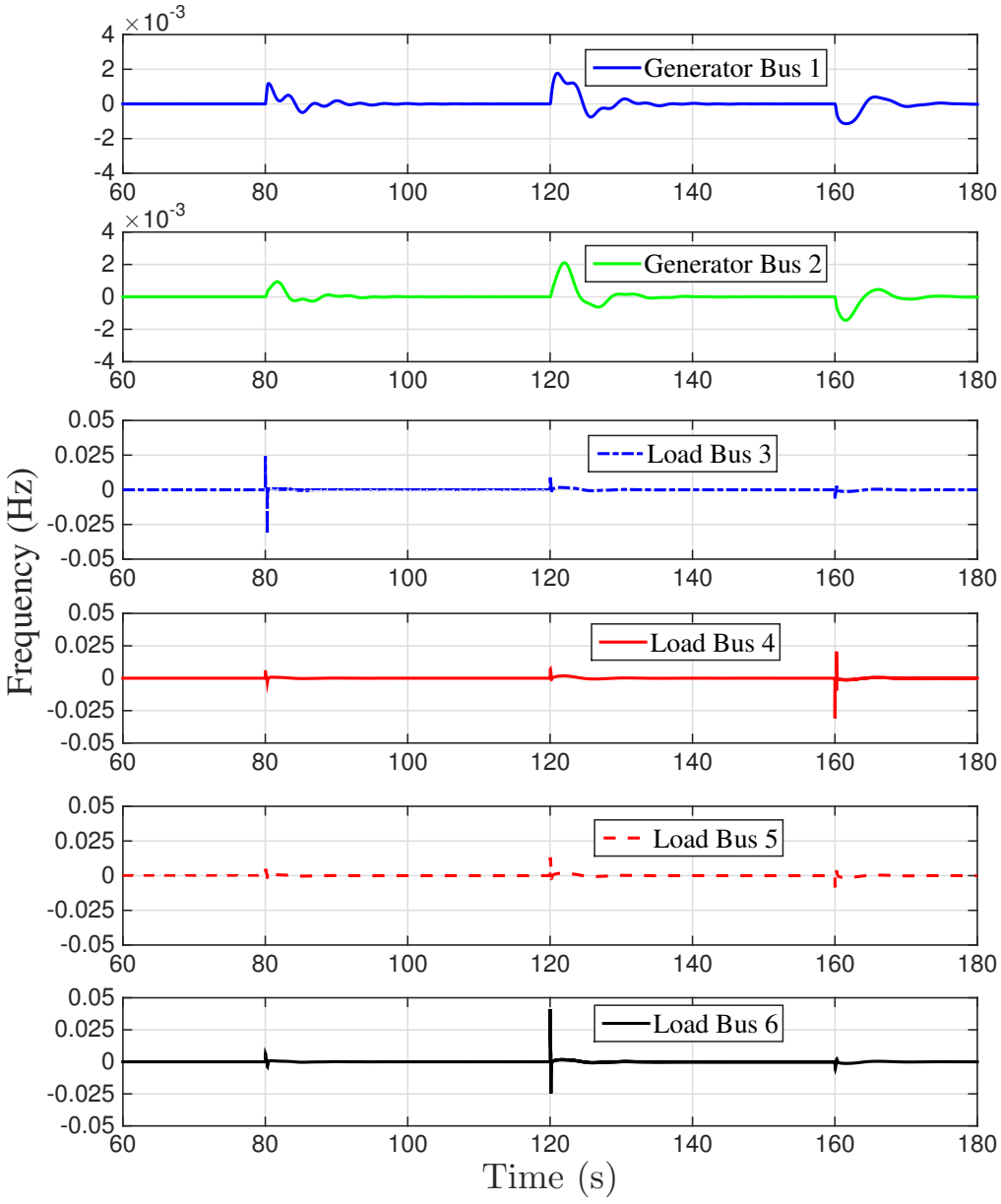


Figure 4.15: Simulation showing the operating frequency deviation of the 6-bus network model. When the reactive power demand at a load bus changes, the operating frequency of the connecting buses fluctuates. The fluctuations are always within the acceptable range.

4.7 Conclusion

This section has considered the modelling of power system networks that contain on-load tap changers. The finite and infinite generator bus model and three different types of tap changer models have been introduced and the combination of three types of tap changer model with different generator bus models have been simulated. The differential algebraic equations of the finite generator bus model with a set of certain parameters were first reformulated as ODEs, making them easier to simulate. In addition, a new continuous time model of an OLTC, that uses a quantiser to enforce the finite number of tap positions, was developed. It was shown that this model could capture the discrete movement of tap change better than the continuous model, whilst being less mathematically challenging than the discrete models as it does not result in a hybrid system. Incorporating the quantisation effect was shown to introduce chattering within the network, which could lead to power fluctuations and OLTC wear. A method to remove this chattering was proposed that introduces "friction" into the system via a deadzone. The method was shown to remove the chattering and have a steady-state error as close as possible to the unquantised equilibrium. Simulations were used to compare the various models considered in the paper. It can be seen that all the tap change modelling techniques has advantages and disadvantages. The aim of this section is to provide an understanding of more physically relevant models of power networks that incorporate tap change.

5 Conclusion

Contents

5.1 Summary and conclusion 147
5.2 Future work 151

Voltage and frequency are key variables in power system, and this thesis concentrates on the detection and analysis of tap changes and the estimation of power system frequency. In this thesis, several work relating to tap changes have been presented. These work are conducted with the purpose of providing more understanding of tap changes and thus to improve the efficiency of power system. In this chapter, the summary and conclusion of this thesis is presented and potential future work is proposed.

5.1 Summary and conclusion

The first chapter is the introduction part for the thesis and sets out the background information for the whole thesis. An overview of the power system and the smart grid is given in this section. The latest research on demand side management, dynamic response are introduced. A brief introduction of OLTC transformer is also included in this chapter.

In the second chapter, an algorithm that can be used for measuring the frequency of the three phase sinusoidal voltage signal of power system is developed. Frequency is one of the most important quantities in power system since it reflects the dynamic energy balance between load and generating power. The utilisation of renewable energy also causes new challenges to the existing power system. As more renewable power generators are being connected to power distribution systems, instability of these systems and of the grid itself can occur. Therefore, it is important to estimate the frequency of power system with high accuracy.

The real voltage signal in a power system, is not a pure sinusoidal wave but contains high harmonics and noise. In order to achieve better frequency estimation performance, we make use of the Prism algorithm and designed an algorithm that combines Prism, PLL and complex least mean square method (CLMS). In this algorithm, we first use Prism algorithm as the pre-filter to eliminate the high harmonic and noise content in the three-phase voltage signal [91]. The output is then processed by a well-designed PLL or Prism which gives an initial estimate of the power system frequency. Finally, the pre-filtered three-phase sinusoidal signal is transformed into complex form single using $\alpha\beta$ transform and put into CLMS with the initial estimate frequency value [83]. The refined estimated value of the three-phase voltage signal can be obtained from the output of CLMS. The simulation results show that the proposed enhanced frequency estimation algorithm track the signal faster and more accurately than the other algorithms that have been considered.

In the third chapter, an algorithm for tap change detection is introduced. This algorithm can detect tap changes from the sinusoidal voltage signal obtained from the low voltage load side. In DSM, an important issue is to provide users with the information of network stress and the stress in a power system can be considered as the gap between the demand and the capacity of the generator [102]. Only when the users know the local network stress information, can they adjust their usage of power according to different load status. Research is being carried out on methods for identifying the local network stress, but most of the stress identifying methods rely

heavily on a dedicated communication network and require communication between users and a control centre, which is both costly and vulnerable. An alternative method that can avoid the costly communication process is to estimate the load status by detecting tap changes. In a power system, OLTC transformers are used to balance the voltage loss along the cable and to keep the voltage at the user side stable. An OLTC transformer can move its tap to different winding contacts, adjusting its transformer ratio and changing its output voltage to react to different load conditions. A tap change can be detected from the user side and can provide information about the variation of load stress, allowing a user to identify network stress without the need for a communication network.

In order to detect the tap changes from voltage fluctuations efficiently, there are three parts in the tap change detection algorithm. The first part is a phase-locked loop (PLL) which can trace the operating frequency of the voltage signal, and a Hilbert-transform phase-locked loop (HTPLL) [94] is used. It was chosen because it could provide better performance when there is distortion in the input voltage signal [96, 143]. The second part is a windowed recursive least square (WRLS) algorithm, which is used for calculating the magnitude of the harmonics within the voltage signal. To ensure numerical robustness, the WRLS algorithm is based on a QR-decomposition WRLS (QRD-WRLS) algorithm [122]. Finally, a hypothesis test is used for detecting the tap changes from the voltage magnitude data obtained from the HTPLL and QRD-WRLS. The generalized likelihood ratio (GLR) test and the two sample t-test [128] can both be used for tap change detection. All three parts of the algorithm are designed to work within sliding windows, so that the algorithm can be implemented for on-line calculation. A set of simulated data and real time data obtained from the real power system are used to test the algorithm. All the tap changes that occurred in the data set are successfully detected. However, some voltage changes were also falsely classified as tap changes. It is also shown that by using appropriate criteria to classify tap changes in the algorithm, the accuracy of the algorithm can be improved.

The fourth chapter focuses on the modelling of tap changer, generator and load buses, which can be used for studying the interaction between tap changes and the variation in load. The operation and control of the modern power grid is layered. Each layer is characterized by its operational objectives, corresponding to its operating time-scale [5, 59]. When studying the secondary control layer, which relates to the dynamics of the load as this normally determines the voltage variation, the tap changer should be modelled. Currently, there are mainly two ways to model tap changers. The first way is to model it as a discrete changing device. In this case, a tap change will be made when the load bus voltage reaches a given limit [133]. The second way is to model the tap changer as a continuous changing device and to ignore the gap between different tap positions resulting in a linear tap changer model can be obtained [135]. Discrete models are able to capture the discrete nature of tap change, but they result in nonlinear hybrid power system models. The control of such nonlinear hybrid systems is known to be challenging, making these models unsuitable for many applications [136]. Accordingly, simpler linear models with continuous dynamics are often used, but these models have been found to give less reliable simulations [59].

In this chapter, we proposed another way for modelling tap changer, which is to combine a linear dynamic function with a quantiser. This proposed model evolves in continuous time but has a quantised number of tap positions. In simulation, it is shown that the use of a quantiser in the new tap changer model leads to chattering problem in the load bus voltage. To overcome this, we designed a deadzone for the model to eliminate the chattering problem. By designing a suitable deadzone, we have shown that the quantisation output for tap changer can always be equal to the required OLTC transformer ratio. The performance of the quantised tap changer model combined with different generator bus model is compared to the continuous and discrete model. Numerical examples show that the output transformer ratio of the proposed model shows discrete changes, which is similar to the performance of real tap changers. The proposed quantised tap changer model is suitable for power system analysis.

5.2 Future work

The following represent suggestions to further develop the results presented in this thesis as part of future research:

- In the third chapter, we have developed a tap change detection algorithm. The tap change information can be used to estimate the local network stress, but how to used the tap change signal to control the variable loads to react to different local network conditions still needs to be investigated. When a OLTC transformer makes a tap change, all downstream users can detect this information, but if they all react to the change without proper regulation, it may lead to a large change in power consumption resulting in a tap change in the opposite direction. Therefore, a suitable distributed control strategy should be designed, so that the tap change information can be properly used.

- In the fourth chapter, we have proposed a modelling technique for OLTC transformer, generators and loads, which can be used to study the interaction between the load and voltage variation. In the model, when considering the loads, we modelled all the load and generator in the same region as single buses. We can extend the study by including renewable energy generator and energy storage device in the system, which will make the model more reliable and applicable.

- According to some recent studies, the interaction between the energy storage device and tap changer can lead to limit cycles [134]. The limit cycle phenomenon caused by the effect of controllable load and tap changer has been studied and simulated in the third chapter. We can extend the study by adding renewable generator and the energy storage device into the model and it will be interesting to investigate the interaction between the tap changer and renewable generators and energy storage devices.

References

[1] International Energy Agency. *Cool Appliances: Policy Strategies for Energy-Efficient Homes*. OECD Publishing, 2003.

[2] V. C. Güngör, D. Sahin, T. Kocak, S. Ergüt, C. Buccella, C. Cecati, and G. P. Hancke. "Smart grid technologies: communication technologies and standards". In: *IEEE transactions on Industrial informatics* 7.4 (2011), pp. 529–539.

[3] N. Gudi, L. Wang, and V. Devabhaktuni. "A demand side management based simulation platform incorporating heuristic optimization for management of household appliances". In: *International Journal of Electrical Power & Energy Systems* 43.1 (2012), pp. 185–193.

[4] D. Rogers and T. Green. "An active-shunt diverter for on-load tap changers". In: *IEEE Transactions on Power Delivery* 28.2 (2013), pp. 649–657.

[5] A. R. Bergen and V. Vittal. *Power Systems Analysis*. Pearson Education India, 2009.

[6] *Power transmission system Kernel Description*. `https://en.wikipedia.org/wiki/Electric_power_transmission`. Accessed: 2015-08-28.

[7] P. Suwanapingkarl. "Power Quality Analysis of Future Power Networks". PhD thesis. Northumbria University, 2012.

[8] John J Grainger and William D Stevenson. *Power System Analysis*. Vol. 621. McGraw-Hill New York, 1994.

[9] P. J. Agrell, P. Bogetoft, and M. Mikkers. "Smart-grid investments, regulation and organization". In: *Energy Policy* 52 (2013), pp. 656–666.

[10] G. Strbac. "Demand side management: Benefits and challenges". In: *Energy Policy* 36.12 (2008), pp. 4419–4426.

[11] P. F. Ribeiro, B. K. Johnson, M. L. Crow, A. Arsoy, and Y. Liu. "Energy storage systems for advanced power applications". In: *Proceedings of the IEEE* 89.12 (2001), pp. 1744–1756.

[12] H. Ibrahim, A. Ilinca, and J. Perron. "Energy storage systems—characteristics and comparisons". In: *Renewable and Sustainable Energy Reviews* 12.5 (2008), pp. 1221–1250.

[13] K. C. Divya and J. Østergaard. "Battery energy storage technology for power systems—An overview". In: *Electric Power Systems Research* 79.4 (2009), pp. 511–520.

[14] P. Mercier, R. Cherkaoui, and A. Oudalov. "Optimizing a battery energy storage system for frequency control application in an isolated power system". In: *IEEE Transactions on Power Systems* 24.3 (2009), pp. 1469–1477.

[15] J. T. Boys, G. A. Covic, and A. W. Green. "Stability and control of inductively coupled power transfer systems". In: *IEEE Proceedings-Electric Power Applications* 147.1 (2000), pp. 37–43.

[16] M. H. Amini, A. Kargarian, and O. Karabasoglu. "ARIMA-based decoupled time series forecasting of electric vehicle charging demand for stochastic power system operation". In: *Electric Power Systems Research* 140 (2016), pp. 378–390.

[17] R. B. Schainker. "Executive overview: energy storage options for a sustainable energy future". In: *IEEE Power Engineering Society General Meeting.* 2004, pp. 2309–2314.

[18] M. H. Nehrir, C. Wang, K. Strunz, H. Aki, R. Ramakumar, J. Bing, Z. Miao, and Z. Salameh. "A review of hybrid renewable/alternative energy systems for electric power generation: Configurations, control, and applications". In: *IEEE Transactions on Sustainable Energy* 2.4 (2011), pp. 392–403.

[19] L. Xu and D. Chen. "Control and operation of a DC microgrid with variable generation and energy storage". In: *IEEE Transactions on Power Delivery* 26.4 (2011), pp. 2513–2522.

[20] J. L. Rodriguez-Amenedo, S. Arnalte, and J. C. Burgos. "Automatic generation control of a wind farm with variable speed wind turbines". In: *IEEE Transactions on energy conversion* 17.2 (2002), pp. 279–284.

[21] C. R. Sarimuthu, V. K. Ramachandaramurthy, K. R. Agileswari, and H. Mokhlis. "A review on voltage control methods using on-load tap changer transformers for networks with renewable energy sources". In: *Renewable and Sustainable Energy Reviews* 62 (2016), pp. 1154–1161.

[22] P. Palensky and D. Dietrich. "Demand side management: Demand response, intelligent energy systems, and smart loads". In: *IEEE transactions on industrial informatics* 7.3 (2011), pp. 381–388.

[23] M. P. Moghaddam, A. Abdollahi, and M. Rashidinejad. "Flexible demand response programs modeling in competitive electricity markets". In: *Applied Energy* 88.9 (2011), pp. 3257–3269.

[24] D. Papadaskalopoulos, R. Moreira, G. Strbac, D. Pudjianto, P. Djapic, F. Teng, and M. Papapetrou. "Quantifying the Potential Economic Benefits of Flexible Industrial Demand in the European Power System". In: *IEEE Transactions on Industrial Informatics* (2018).

[25] H. Morais, T. Sousa, Z. Vale, and P. Faria. "Evaluation of the electric vehicle impact in the power demand curve in a smart grid environment". In: *Energy Conversion and Management* 82 (2014), pp. 268–282.

[26] B. Hayes, I. Hernando-Gil, A. Collin, G. Harrison, and S. Djokic. "Optimal power flow for maximizing network benefits from demand-side management". In: *IEEE Transactions on Power systems* 29.4 (2014), pp. 1739–1747.

[27] N. Ruiz, Iñ. Cobelo, and J. Oyarzabal. "A direct load control model for virtual power plant management". In: *IEEE Transactions on Power Systems* 24.2 (2009), pp. 959–996.

[28] L. Gelazanskas and K. Gamage. "Demand side management in smart grid: A review and proposals for future direction". In: *Sustainable Cities and Society* 11 (2014), pp. 22–30.

[29] L. Bird, M. R. Milligan, and D. Lew. *Integrating Variable Renewable Energy: Challenges and Solutions*. National Renewable Energy Laboratory, 2013.

[30] P. Zhang, Y. Yang, Y. Zheng, L. Wang, and X. Li. "Opportunities and challenges for renewable energy policy in China". In: *Renewable and Sustainable Energy Reviews* 13.2 (2009), pp. 439–449.

[31] Clark W Gellings. "The concept of demand-side management for electric utilities". In: *Proceedings of the IEEE* 73.10 (1985), pp. 1468–1470.

[32] A. Arteconi, N. J. Hewitt, and F. Polonara. "State of the art of thermal storage for demand-side management". In: *Applied Energy* 93 (2012), pp. 371–389.

[33] I. Atzeni, L. G. Ordóñez, G. Scutari, D. P. Palomar, and J. R. Fonollosa. "Demand-side management via distributed energy generation and storage optimization". In: *IEEE Transactions on Smart Grid* 4.2 (2013), pp. 866–876.

[34] P. Warren. "A review of demand-side management policy in the UK". In: *Renewable and Sustainable Energy Reviews* 29 (2014), pp. 941–951.

[35] A. Arteconi and F. Polonara. "Assessing the Demand Side Management Potential and the Energy Flexibility of Heat Pumps in Buildings". In: *Energies* 11.7 (2018), p. 1846.

[36] K. Ng and G. B. Sheble. "Direct load control-a profit-based load management using linear programming". In: *IEEE Transactions on Power Systems* 13.2 (1998), pp. 688–694.

[37] R. de Sá Ferreira, L. A. Barroso, P. Rochinha Lino, M. M. Carvalho, and P. Valenzuela. "Time-of-use tariff design under uncertainty in price-elasticities of electricity demand: A stochastic optimization approach". In: *IEEE Transactions on Smart Grid* 4.4 (2013), pp. 2285–2295.

[38] E. Celebi and J. D. Fuller. "A model for efficient consumer pricing schemes in electricity markets". In: *IEEE Transactions on Power Systems* 22.1 (2007), pp. 60–67.

[39] S. Datchanamoorthy, S. Kumar, Y. Ozturk, and G. Lee. "Optimal time-of-use pricing for residential load control". In: *2011 IEEE International Conference on Smart Grid Communications (SmartGridComm)*. IEEE. Brussels, Belgium, 2011, pp. 375–380.

[40] X. Fang, S. Misra, G. Xue, and D. Yang. "Smart grid—The new and improved power grid: A survey". In: *IEEE Communications Surveys & Tutorials* 14.4 (2012), pp. 944–980.

[41] Q. Shen, B. Ramachandran, S. K. Srivastava, M. Andrus, and D. A. Cartes. "Power and energy management in integrated power system". In: *IEEE Electric Ship Technologies Symposium (ESTS)*. IEEE. 2011, pp. 414–419.

[42] P. Kumar and D. P. Kothari. "Recent philosophies of automatic generation control strategies in power systems". In: *IEEE transactions on power systems* 20.1 (2005), pp. 346–357.

[43] N. Jaleeli, L. S. VanSlyck, D. N. Ewart, L. H. Fink, and A. G. Hoffmann. "Understanding automatic generation control". In: *IEEE transactions on power systems* 7.3 (1992), pp. 1106–1122.

[44] D. D. Siljak, D. M. Stipanovic, and A. I. Zecevic. "Robust decentralized turbine/governor control using linear matrix inequalities". In: *IEEE Transactions on Power Systems* 17.3 (2002), pp. 715–722.

[45] M. Beus and H. Pandžić. "Application of model predictive control algorithm on a hydro turbine governor control". In: *Power Systems Computation Conference (PSCC)*. IEEE. 2018.

[46] W. W. Hogan. "Contract networks for electric power transmission". In: *Journal of regulatory economics*. Vol. 4. 2. Springer, 1992, pp. 211–242.

[47] W. El-Khattam, K. Bhattacharya, Y. Hegazy, and M. Salama. "Optimal investment planning for distributed generation in a competitive electricity market". In: *IEEE Transactions on power systems* 19.3 (2004), pp. 1674–1684.

[48] D. C. Walters and G. B. Sheble. "Genetic algorithm solution of economic dispatch with valve point loading". In: *IEEE transactions on Power Systems* 8.3 (1993), pp. 1325–1332.

[49] P. Chen and H. Chang. "Large-scale economic dispatch by genetic algorithm". In: *IEEE transactions on power systems* 10.4 (1995), pp. 1919–1926.

[50] J. Hetzer, C. Y. David, and K. Bhattarai. "An economic dispatch model incorporating wind power". In: *IEEE Transactions on energy conversion* 23.2 (2008), pp. 603–611.

[51] X. Zhang. "Scalable (Re)design Frameworks for Optimal, Distributed Control in Power Networks". PhD thesis. University of Oxford, 2015.

[52] P. Kundur, N. J. Balu, and M. G. Lauby. *Power System Stability and Control*. McGraw-hill New York, 1994.

[53] J. W. Chapman, M. D. Ilic, C. A. King, L. Eng, and H. Kaufman. "Stabilizing a multimachine power system via decentralized feedback linearizing excitation control". In: *IEEE Transactions on Power Systems* 8.3 (1993), pp. 830–839.

[54] C. A. King, J. W. Chapman, and M. D. Ilic. "Feedback linearizing excitation control on a full-scale power system model". In: *IEEE Transactions on Power systems* 9.2 (1994), pp. 1102–1109.

[55] C. Gao and M. A. Redfern. "A review of voltage control techniques of networks with distributed generations using On-Load Tap Changer transformers". In: *Proceedings of the 45th International Universities Power Engineering Conference*. IEEE. Cardiff, 2010.

[56] E. F. Mogos and X. Guillaud. "A voltage regulation system for distributed generation". In: *IEEE Power Systems Conference and Exposition*. IEEE. 2004, pp. 787–794.

[57] M. A. Kashem and G. Ledwich. "Distributed generation as voltage support for single wire earth return systems". In: *IEEE Transactions on Power Delivery* 19.3 (2004), pp. 1002–1011.

[58] C. Oates, A. Barlow, and V. Levi. "Tap changer for distributed power". In: *European Conference on Power Electronics and Applications*. IEEE. 2007.

[59] X. Zhang, R. Kang, M. McCulloch, and A. Papachristodoulou. "Real-time active and reactive power regulation in power systems with tap-changing transformers and controllable loads". In: *Sustainable Energy, Grids and Networks* 5 (2016), pp. 27–38.

[60] J. de Oliveira Quevedo, F. E. Cazakevicius, R. C. Beltrame, T. B. Marchesan, L. Michels, C. Rech, and L. Schuch. "Analysis and design of an electronic on-load tap changer distribution transformer for automatic voltage regulation". In: *IEEE Transactions on Industrial Electronics* 64.1 (894 2017), p. 883.

[61] F. A. Viawan, A. Sannino, and J. Daalder. "Voltage control with on-load tap changers in medium voltage feeders in presence of distributed generation". In: *Electric Power Systems Research* 77.10 (2007), pp. 1314–1322.

[62] M. Bahadornejad and N. C. Nair. "Intelligent control of on-load tap changing transformer". In: *IEEE Transactions on Smart Grid* 5.5 (2014), pp. 2255–2263.

[63] R. Echavarria, A. Claudio, and M. Cotorogea. "Analysis, design, and implementation of a fast on-load tap changing regulator". In: *IEEE Transactions on Power Electronics* 22.2 (2007), pp. 527–534.

[64] C. Long and L. F. Ochoa. "Voltage control of PV-rich LV networks: OLTC-fitted transformer and capacitor banks". In: *IEEE Transaction on Power Systems* 31.5 (2016), pp. 4016–4025.

[65] N. Daratha, B. Das, and J. Sharma. "Coordination between OLTC and SVC for voltage regulation in unbalanced distribution system distributed generation". In: *IEEE Transactions on Power Systems* 29.1 (2014), pp. 289–299.

[66] P. Cartwright, L. Holdsworth, J. B. Ekanayake, and N. Jenkins. "Co-ordinated voltage control strategy for a doubly-fed induction generator (DFIG)-based wind farm". In: *IEE Proceedings-Generation, Transmission and Distribution* 151.4 (2004), pp. 495–502.

[67] M. P. Henry. "An introduction to Prism signal processing applied to sensor validation". In: *Measurement Techniques* (2018), pp. 1–5.

[68] X. Chen and S. R. Duncan. "Automatic detection of tap changes on an electricity grid". In: *Proceedings of the UKACC 11th International Conference on Control*. IEEE. Belfast, 2016.

[69] S. D. Ramchurn, P. Vytelingum, A. Rogers, and N. Jennings. "Agent-based control for decentralised demand side management in the smart grid". In: *The 10th International Conference on Autonomous Agents and Multiagent Systems*. Vol. 1. International Foundation for Autonomous Agents and Multiagent Systems. 2011, pp. 5–12.

[70] K. R. Padiyar. *Power System Dynamics*. BS publications, 2008.

[71] D. A. Gonzalez and J. C. McCall. "Design of filters to reduce harmonic distortion in industrial power systems". In: *IEEE Transactions on Industry Applications* 3 (1987), pp. 504–511.

[72] M. M. Begovic, P. M. Djuric, S. Dunlap, and A. G. Phadke. "Frequency tracking in power networks in the presence of harmonics". In: *IEEE Transactions on Power Delivery* 8.2 (1993), pp. 480–486.

[73] M. Li and M. P. Henry. "Signal processing methods for Coriolis Mass Flow Metering in two-phase flow conditions". In: *IEEE International Conference on Industrial Technology, Taipei*. IEEE. 2016, pp. 690–695.

[74] Y. Xia, Y. He, K. Wang, W. Pei, Z. Blazic, and D. P. Mandic. "A complex least squares enhanced smart DFT technique for power system frequency estimation". In: *IEEE Transactions on Power Delivery* 32.3 (2017), pp. 1270–1278.

[75] J. K. Hwang and P. N. Markham. "Power system frequency estimation by reduction of noise using three digital filters". In: *IEEE Transactions on Instrumentation and Measurement* 63.2 (2014), pp. 402–409.

[76] M. Karimi-Ghartemani, B. Ooi, and A. Bakhshai. "Application of enhanced phase-locked loop system to the computation of synchrophasors". In: *IEEE Transactions on Power Delivery* 26.1 (2011), pp. 22–32.

[77] B. Wu and M. Bodson. "A magnitude/phase-locked loop approach to parameter estimation of periodic signals". In: *IEEE Transactions on Automatic Control* 48.4 (2003), pp. 612–618.

[78] A. Patapoutian. "On phase-locked loops and Kalman filters". In: *IEEE Transactions on Communications* 47.5 (1999), pp. 670–672.

[79] P. K. Dash, A. K. Pradhan, and G. Panda. "Frequency estimation of distorted power system signals using extended complex Kalman filter". In: *IEEE Transactions on Power Delivery* 14.3 (1999), pp. 761–766.

[80] A. Routray, A. K. Pradhan, and K. P. Rao. "A novel Kalman filter for frequency estimation of distorted signals in power systems". In: *IEEE Transactions on Instrumentation and Measurement* 51.3 (2002), pp. 469–479.

[81] P. K. Dash, G. Panda, A. K. Pradhan, A. Routray, and B. Duttagupta. "An extended complex Kalman filter for frequency measurement of distorted signals". In: *IEEE Power Engineering Society Winter Meeting*. Vol. 3. IEEE. 2000, pp. 1569–1574.

[82] B. F. La Scala and R. R. Bitmead. "Design of an extended Kalman filter frequency tracker". In: *IEEE Transactions on Signal Processing* 44.3 (1996), pp. 739–742.

[83] A. K. Pradhan, A. Routray, and A. Basak. "Power system frequency estimation using least mean square technique". In: *IEEE transactions on power delivery* 20.3 (2005), pp. 1812–1816.

[84] M. D. Kusljevic, J. J. Tomic, and L. D. Jovanovic. "Frequency estimation of three-phase power system using weighted-least-square algorithm and adaptive FIR filtering". In: *IEEE Transactions on Instrumentation and Measurement* 59.2 (2010), pp. 322–329.

[85] T. C. Hsia. *System Identification: Least-Squares Methods*. Lexington books Lexington, 1977.

[86] A. Choi. "Real-time fundamental frequency estimation by least-square fitting". In: *IEEE Transactions on Speech and Audio Processing* 5.2 (1997), pp. 201–205.

[87] T. Lobos, T. Kozina, and H. Koglin. "Power system harmonics estimation using linear least squares method and SVD". In: *IEE Proceedings-Generation, Transmission and Distribution* 148.6 (2001), pp. 567–572.

[88] A. G. Phadke. "Synchronized phasor measurements in power systems". In: *IEEE Computer Applications in power* 6.2 (1993), pp. 10–15.

[89] H. Karimi, M. Karimi-Ghartemani, and M. R. Iravani. "Estimation of frequency and its rate of change for applications in power systems". In: *IEEE Transactions on Power Delivery* 19.2 (2004), pp. 472–480.

[90] C. Huang, C. Lee, K. Shih, and Y. Wang. "Frequency estimation of distorted power system signals using a robust algorithm". In: *IEEE Transactions on Power Delivery* 23.1 (2008), pp. 41–51.

[91] M. Henry, F. Leach, M. Davy, O. Bushuev, M. Tombs, F. Zhou, and S. Karout. "The Prism: efficient signal processing for the internet of things". In: *IEEE Industrial Electronics Magazine* 11.4 (2017), pp. 22–32.

[92] W. C. Lindsey and M. K. Simon. *Phase-Locked Loops & Their Application*. New York : IEEE Press, 1978.

[93] R. E. Best. *Phase-Locked Loops : Theory, Design, and Applications*. New York : McGraw-Hill, 1984.

[94] S. C. Gupta. "Phase-locked loops". In: *Proceedings of the IEEE* 63.2 (1975), pp. 291–306.

[95] J. B. Encinas. *Phase Locked Loops*. London : Chapman & Hall, 1993.

[96] D. W. Clarke. "Designing phase-locked loops for instrumentation applications". In: *Measurement* 32.3 (2002), pp. 205–227.

[97] J. C. Goswami and A. E. Hoefel. "Algorithms for estimating instantaneous frequency". In: *Signal Processing* 84.8 (2004), pp. 1423–1427.

[98] A. V. Oppenheim and R. W. Schafer. *Digital Signal Processing*. London : Prentice-Hall International, 1986.

[99] M. S. Sachdev and M. M. Giray. "A least error squares technique for determining power system frequency". In: *IEEE Transactions on Power Apparatus and Systems* 2 (1985), pp. 437–444.

[100] B. Widrow, J. McCool, and M. Ball. "The complex LMS algorithm". In: *Proceedings of the IEEE* 63.4 (1975), pp. 719–720.

[101] D. Geelen, A. Reinders, and D. Keyson. "Empowering the end-user in smart grids: Recommendations for the design of products and services". In: *Energy Policy* 61 (2013), pp. 151–161.

[102] R. Kang and M. McCulloch. "Identification of electric power system stress through feeder voltage measurement". In: *Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), 2014 IEEE PES*. IEEE. Istanbul, Turkey, 2014, pp. 1–6.

[103] H. Bevrani. *Robust Power System Frequency Control*. Springer, 2009.

[104] J. A. Short, D. G. Infield, and L. L. Freris. "Stabilization of grid frequency through dynamic demand control". In: *IEEE Transactions on power systems* 22.3 (2007), pp. 1284–1293.

[105] Z. Fan, P. Kulkarni, S. Gormus, C. Efthymiou, G. Kalogridis, M. Sooriyabandara, Z. Zhu, S. Lambotharan, and W. H. Chin. "Smart grid communications: overview of research challenges, solutions, and standardization activities". In: *Communications Surveys & Tutorials, IEEE* 15.1 (2013), pp. 21–38.

[106] M. Qasim, P. Kanjiya, and V. Khadkikar. "Artificial-neural-network-based phase-locking scheme for active power filters". In: *IEEE Transactions on Industrial Electronics* 61.8 (2014), pp. 3857–3866.

[107] R. Ravichandran, E. Saba, K. Chen, M. Goel, S. Gupta, and S. N. Patel. "WiBreathe: Estimating respiration rate using wireless signals in natural settings in the home". In: *IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE. 2015, pp. 131–139.

[108] H. Wen, S. Guo, Z. Teng, F. Li, and Y. Yang. "Frequency estimation of distorted and noisy signals in power systems by FFT-based approach". In: *IEEE Transactions on Power Systems* 29.2 (2014), pp. 765–774.

[109] J. Borkowski, D. Kania, and J. Mroczka. "Interpolated-DFT-based fast and accurate frequency estimation for the control of power". In: *IEEE Transactions on industrial Electronics* 61.12 (2014), pp. 7026–7034.

[110] S. V. Hareesh, P. Raja, and M. P. Selvan. "An effective implementation of Phasor measurement unit (PMU) by using non-recursive DFT algorithm". In: *International Conference on Condition Assessment Techniques in Electrical Systems (CATCON)*. IEEE. 2015, pp. 195–199.

[111] A. Derviškadić, P. Romano, and M. Paolone. "Iterative-Interpolated DFT for Synchrophasor Estimation: A Single Algorithm for P-and M-Class Compliant PMUs". In: *IEEE Transactions on Instrumentation and Measurement* 67.3 (2018), pp. 547–558.

[112] S. Golestan, J. M. Guerrero, and J. C. Vasquez. "Three-phase PLLs: A review of recent advances". In: *IEEE Transactions on Power Electronics* 32.3 (2017), pp. 1894–1907.

[113] F. Messina, P. Marchi, L. R. Vega, C. G. Galarza, and Hé. Laiz. "A novel modular positive-sequence synchrophasor estimation algorithm for PMUs". In: *IEEE Transactions on Instrumentation and Measurement* 66.6 (2017), pp. 1164–1175.

[114] T. Söderström and P. Stoica. *System Identification*. New York ; London : Prentice-Hall, 1989.

[115] M. R. Bai, J. Jeng, and C. Chen. "Adaptive order tracking technique using recursive least-square algorithm". In: *Journal of vibration and acoustics* 124.4 (2002), pp. 502–511.

[116] B. Choi and Z. Bien. "Sliding-windowed weighted recursive least-squares method for parameter estimation". In: *Electronics Letters* 25.20 (1989), pp. 1381–1382.

[117] J. Cioffi and T. Kailath. "Fast, recursive-least-squares transversal filters for adaptive filtering". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 32.2 (1984), pp. 304–337.

[118] K. Zhao, F. Ling, H. Lev-Ari, and J. G. Proakis. "Sliding window order-recursive least-squares algorithms". In: *IEEE transactions on signal processing* 42.8 (1994), pp. 1961–1972.

[119] S. K. Singh, N. Sinha, A. K. Goswami, and N. Sinha. "Robust estimation of power system harmonics using a hybrid firefly based recursive least square algorithm". In: *International Journal of Electrical Power and Energy Systems* 80 (2016), pp. 287–296.

[120] P. Strobach and D. Goryn. "A computation of the sliding window recursive QR decomposition". In: *IEEE International Conference on Acoustics, Speech, and Signal Processing, 1993. ICASSP-93.* Vol. 4. IEEE. Minneapolis, Minnesota, USA, 1993, pp. 29–32.

[121] H. Sakai and H. Nakaoka. "A fast sliding window QRD-RLS algorithm". In: *Signal processing* 78.3 (1999), pp. 309–319.

[122] K. Yoo and H. Park. "Fast residual computation for sliding window recursive least squares methods". In: *Signal Processing* 45.1 (1995), pp. 85–95.

[123] G. H. Golub and C. F. Van Loan. *Matrix Computations*. 2nd. Baltimore : The Johns Hopkins University Press, 2013.

[124] R. H. Fowler and L. Cohen. *Statistic*. Cambridge University Press London, 1955.

[125] Dengsheng Lu, P Mausel, E Brondizio, and Emilio Moran. "Change detection techniques". In: *International journal of remote sensing* 25.12 (2004), pp. 2365–2401.

[126] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam. "Image change detection algorithms: a systematic survey". In: *IEEE transactions on image processing* 14.3 (2005), pp. 294–307.

[127] A. Willsky and H. Jones. "A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems". In: *IEEE Transactions on Automatic control* 21.1 (1976), pp. 108–112.

[128] K. Koch. *Parameter Estimation and Hypothesis Testing in Linear Models*. Berlin : Springer-Verlag, 1988.

[129] F. Gustafsson. *Adaptive Filtering and Change Detection*. Chichester : Wiley, 2000.

[130] F. Gustafsson. "The marginalized likelihood ratio test for detecting abrupt changes". In: *IEEE Transactions on automatic control* 41.1 (1996), pp. 66–78.

[131] D. Taeger. *Statistical Hypothesis Testing with SAS and R*. Hoboken, New Jersey : Wiley, 2014.

[132] N. Cressie and H. Whitford. "How to use the two sample t-test". In: *Biometrical Journal* 28.2 (1986), pp. 131–148.

[133] Q. Wu, D. H. Popović, and D. J. Hill. "Avoiding sustained oscillations in power systems with tap changing transformers". In: *International Journal of Electrical Power & Energy Systems* 22.8 (2000), pp. 597–605.

[134] D. Ranamuka, A. P. Agalgaonkar, K. M. Muttaqi, and M. E. Alam. "Mitigating tap changer limit cycles in modern electricity networks embedded with local generation units". In: *IEEE Transactions on Industry Applications* 52.1 (2016), pp. 455–465.

[135] C. C. Liu and K. T. Vu. "Analysis of tap-changer dynamics and construction of voltage stability regions". In: *IEEE Transactions on Circuits and Systems* 36.4 (1989), pp. 575–590.

[136] I. A. Hiskens and M. A. Pai. "Hybrid systems view of power system modelling". In: *Proceedings of the The IEEE International Symposium on Circuits and Systems (ISCAS)*. Vol. 2. IEEE. Geneva, 2000, pp. 228–231.

[137] X. Zhang and A. Papachristodoulou. "A real-time control framework for smart power networks with star topology". In: *Proceedings of the American Control Conference*. IEEE. Washington, DC, 2013, pp. 5062–5067.

[138] D. J. Hill, I. A. Hiskens, and D. H. Popovic. "Stability analysis of power system loads with recovery dynamics". In: *International Journal of Electrical Power & Energy Systems* 16.4 (1994), pp. 277–286.

[139] D. J. Hill. "Nonlinear dynamic load models with recovery for voltage stability studies". In: *IEEE Transactions on Power Systems* 8.1 (1993), pp. 166–176.

[140] M. Larsson, D. H. Popović, and D. J. Hill. "Limit cycles in power systems due to OLTC deadbands and load–voltage dynamics". In: *Electric Power Systems Research* 47.3 (1998), pp. 181–188.

[141] J. Medanic, M. Ilic-Spong, and J. Christensen. "Discrete models of slow voltage dynamics for under load tap-changing transformer coordination". In: *IEEE Transactions on Power Systems* 2.4 (1987), pp. 873–880.

[142] K. H. Johansson, A. Rantzer, and K. J. Åström. "Fast switches in relay feedback systems". In: *Automatica* 35.4 (1999), pp. 539–552.

[143] A. V. Oppenheim. *Discrete-Time Signal Processing*. Pearson Education India, 1999.